Jakub Leszek Pach
https://orcid.org/0000-0002-3483-4340

IDENTIFYING SAMPLES OF HISTORICAL HANDWRITING VIA IMAGE PROCESSING WITH THE HELP OF CHARACTERISTIC FEATURES OF THE AUTHOR'S HANDWRITING

DOI: 10.36155/PLib.8.00002

The written word marks the boundary between prehistory and history. The invention of writing, which thrust the history of humanity into a new era of civilisation, meant that ideas that had previously only been transmitted orally – how to start a fire, how to make tools: ideas that were essential for civilisation – could be saved from oblivion. These ideas could now be recorded and thus remembered for longer than was possible by relying simply on human memory.

All written texts, from those using pictographic and ideographic writing systems to those using phonetic systems, are material products. They have been created by the human hand. The owner of these written texts is not always the writer, of course: not infrequently, as we know from history, a single mind has been able to control many hands. An example of this is the medieval scribes who, right across Europe, copied sacred works on behalf of the Catholic Church.

As Aristotle said[1], writing – or, more precisely, the essence or „substance" of writing – has both form and matter. Its form is the letters left behind by the hand that wrote it and that carries the biomarker. These may be in a particular script, of course, such as Carolingian miniscule, bastarda or antiqua, but they will always contain an indelible trace of the specific writer who made the forms themselves. The substance, of course, is the contents of the text. The basic task of analysing the text is therefore to establish the authenticity and authorship of the document and its contents.

The first scientific discipline to take these questions as the subject of its research was palaeography. The word itself derives from the Greek *palaiós* „old" and *graphō* „to write", so it is literally the study of „old writing". The discipline of palaeography was born in the second half of the seventeenth century; its beginnings are attributed to the Jesuit Daniel Van Paperbroeck and the Benedictine monk Jean Mabillon.[2] However, it is worth noting that the earliest documented analysis of historical texts dates back to the times of the historian Herodotus in the late fourth or early fifth century B.C.[3] The aim of that discipline was to analyse texts (including questions of their authenticity and distortion), catalogue them and archive them. But it also aimed to study in a broad sense the development of writing as a historical process, from the majuscules of the early Middle Ages to the minuscule of the first incunables. Palaeography also gave rise to another scientific field: neography. As the name suggests, neography is the science of modern writing and is an extension of palaeography. Both fields are considered auxiliary sciences of history.

The problem of identifying a text in the partial or complete absence of information about its provenance or author, and questions of forgeries and falsification, concerned even by the earliest researchers. The tools at their disposal up until the beginning of

1 R.J. Hankinson, *Cause and explanation in ancient Greek thought*, Austin 2001, pp. 96–100.

2 J. Szymański, *Nauki pomocnicze historii*, 6th ed., Warszawa 2001.

3 Herodotus, *The Histories*, transl. Robin Waterfield, Oxford 1997.

the twenty-first century were the „wiseman's glass and eye", the skills of the historian, and deductive, inductive and abductive reasoning. Information technology was still in its infancy. Information about the time and place of the origin of a text could be deduced from its contents, and thanks to the shape of the letters, the circle of possible authors could be narrowed down. Still, not infrequently the work had to be assigned to Gallus Anonymous, for example, or some other "pseudo-author".

All over the world, archive materials and works of art, including manuscripts, paintings and sculptures from all periods – antiquity, the Middle Ages, the Enlightenment – have been lost, stolen or damaged in the turmoil of war, epidemics and natural disasters. When they did manage to make it back to their homes, they were often damaged, had parts missing or were now entirely anonymous. Yet the value of these items, in simplistic terms, depended on two parameters: the hand that made them, and their contents. For example, Rembrandt's "Self-portrait in a Cap"[4] from 1630 and his „Beggar Seated Warming His Hands at a Chafing Dish"[5] from the same year are worth astronomical amounts in the world of culture due to the sum of Rembrandt's achievements during his lifetime and his fame. If their authorship were unknown, a non-expert would have to class them as anonymous and therefore worthless. Indeed, it is safe to say that historical materials exist in collections all over the world which, due to the impossibility of identifying them, remain nameless and, in a certain sense, worthless.

Working with written texts, including identifying them by hand and determining their authenticity, is the speciality of the graphologist. It requires expertise and is expensive and laborious. A graphologist, while performing an analysis of a manuscript, uses another sample of handwriting, and through analysing the

4 Rembrandt van Rijn, *Self-portrait in a Cap, Wide-eyed and Open-mouthed*, 1630, Scientific Library of thePAAS and the PAS in Cracow, Graphic Collections Department, BGR.008347.

5 Rembrandt van Rijn, *Beggar Seated Warming His Hands at a Chafing Dish*, 1630, Scientific Library of thePAAS and the PAS in Cracow, Graphic Collections Department, BGR. 004086.

same letters in both sources, they determine whether they have been written by the same person. In addition, the human factor must be taken into account. Few archives, libraries and scientific institutions can afford the services of a graphologist. Here, however, as Thomas à Kempis wrote – a quote often cited by C.S. Lewis – "the highest does not stand without the lowest". Palaeography, neography and their legacy have given rise to various disciplines within information technology that deal with the analysis of written texts, including pattern recognition, computer vision and template matching. Thanks to the huge development of electronics at the turn of the century, computer science offers many possibilities, including: the recognition of printed texts (commonly known as "optical character recognition" or OCR), not just from scans but from digital photographs; the identification of pieces of music from fragments lasting just a few seconds; the identification of human faces by digital cameras in real time; and free navigation for cars using any modern smartphone.

Although the possibilities offered by machine-learning are growing constantly, they have not yet solved all our problems. In one area, however, they have made major progress. In the second decade of the twenty-first century, the field of pattern recognition has been increasingly successful in solving complex problems which, just a few years ago, were impossible due to insufficient computing power and the use of an overly complex mathematical model. Today, most of these problems can be solved in real time with the help of deep neural networks.

The digital analysis of written texts can take place either offline or online. The offline, or traditional approach involves the analysis of texts written on papyrus, parchment or paper. There is no contact with the author of the document, only with what they have written. The more modern, online approach involves the use of electronic devices – tablets, touchscreens, styluses, digital pens – to record not only the contents but also the pressure exerted during the writing process, the trajectory of the writing instrument, the writing time and other parameters that traditional analysis does not take into account. In this study we are interested in the digital representation of historical manuscripts, so we use offline analysis to try to recognise the contents of texts. However, we are also interested in a second aspect, namely that is, identifying the handwriting in the sample.

One should take into account another aspect – the static analysis of writing distinguishes two methods of analysis: through matter and form and through form only. In the first approach, we analyse the content of the text, divide it into particular letters, and then we analyse the letters in both samples to discover differences and similarities. To do so, the machine needs to be able to read the content of the samples before it analyses them. In the second approach only the form is used to retrieve the author's biomarker and then the mathematical models of both samples are analysed.

The aim of this article is thus to introduce the reader to the subject of digital manuscript analysis. We begin with a theoretical description of a prototype for a system for identifying handwriting samples; this section is based on the author's doctoral thesis,[6] written at Warsaw University of Technology and defended in 2019. Warsaw University of Technology is the only institution in Poland dealing with the recognition and identification of historical handwriting samples using image processing methods. We follow the theoretical description with a description of what the system can do. The system allows us to analyse similarities between writing samples from a micro scale – strokes (components of letters), words and paragraphs – to a macro scale, that is entire manuscripts, including codex collections. When tested with a collection of Latin codices from the Middle Ages from the digital repository Polona.pl, owned by the National Library of Poland,[7] the system correctly identified the authors of handwriting samples almost a hundred per cent of the time. The success rate was similar for an analysis of the database of contemporary manuscripts in English of the

6 J.L. Pach, *Identyfikacja autora rękopisu łacińskiego z wykorzystaniem metod przetwarzania obrazów*, unpublished doctoral thesis, Warsaw University of Technology, 2018.

7 MS 12511 II, MS 3307 II, MS BOZ 36, 1197 V MS.

Computer Science Institute at the University of Bern (the Modified database of the Institute of Informatics and Applied Mathematics, or MIAM),[8] part of research carried out to demonstrate the potential of the prototype.[9] The system was then successfully used to analyse the text of the seventeenth-century Chronicle of Father Stefan Ranatowicz,[10] which demonstrate his calligraphic artistry, and to identify samples of writing by two other authors who added to the Chronicle after his death, namely Father Michał Aquilin Gorczyński and Father Krzysztof Piasecki.[11]

The architecture of the new system is shown in Figure 1. It consists of four main stages: *Pre-processing*, *Text segmentation*, *Extraction of handwriting features* and *Classification*. Comparing these four stages to text analysis, we could say that the first two stages are like the introduction, the *Extraction of handwriting features* is like the development, and the *Classification* is like the conclusion.

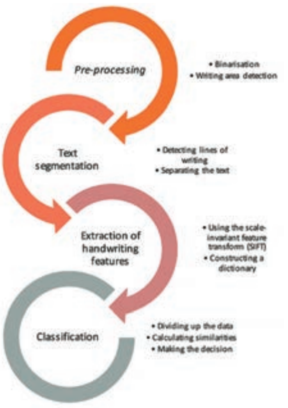


Fig. 1. Architecture of the system for identifying samples of historical handwriting via image processing with the help of characteristic features of the author's handwriting.

8 U.-V. Marti, H. Bunke, „The IAM-database: an English sentence database for offline handwriting recognition", *International Journal on Document Analysis and Recognition*, 2002, 5/1, pp. 39–46.

9 This base is available: Research Group on Computer Vision and Artificial Intelligence – Computer Vision and Artificial Intelligence (heia-fr.ch) [accessed 04.12.2020].

10 S. Ranatowicz, Casimiriae civitatis, MS BJ 3742 III.

11 K. Łatak, M. Pęgier, J. Pach, *Z problematyki kodykologicznej i paleograficznej Kroniki Stefana Ranatowicza*, Warszawa 2019.

Before we move on to the first stage, we should explain briefly what is meant by a „digital image” in “computer science. Historians, in their work, draw on the phenomenon of „subtractive synthesis”, while computer scientists draw more on „additive synthesis”. Both refer to the phenomenon of colour mixing (see Figure 2). Subtractive synthesis occurs when paints or inks are mixed; it is used in today’s printing technology. Thus, to get green ink, you mix blue and yellow. In theory it should be possible to obtain black paint by mixing all the different-coloured paints together. In practice, however, the black colour in printing is placed in a separate container in order not to use up the coloured inks unnecessarily. Additive synthesis is based on the theory of visible light. The colour white contains all the primary colours. This can be seen by splitting visible light with a prism, for example, or in the natural phenomenon of a rainbow.

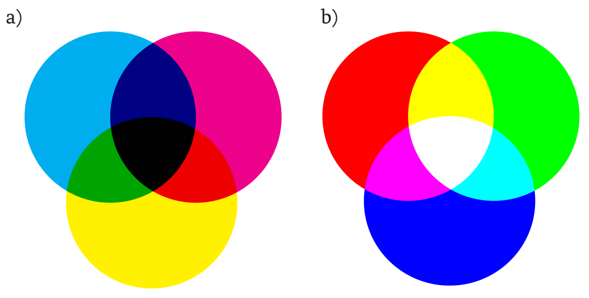


Fig. 2. a) Subtractive synthesis; b) Additive synthesis.

In the twenty-first century, additive synthesis is replacing subtractive synthesis. It is used in all electronic devices that emit visible light, such as monitors, smartphones, TV sets and tablets. These devices emit beams of red, green and blue light, which at maximum intensity together create white light. In fact, many “colour spaces” are used in recording digital images, but here we will only refer to those commonly used in additive synthesis, namely red, green and blue, or the “RGB colour space”. In order for one pixel on the screen to be filled with colour, three numbers are needed, describing the percentage of intensity from zero to 100 per cent of each the three components (red, green and blue). Every colour can be obtained by combining these three numbers. The entire digital image or "raster image" made up of pixels is represented on the computer in the form of a single, two-dimensional matrix for each component. This, in simple terms, is the form in which images are stored in BMP, JPG and other file types.

Additive synthesis has one major advantage when it comes to image analysis. The colour white is a number other than zero, while the colour black is the absence of light, and hence zero. This means that everything other than zero can be manipulated and transformed (whereas multiplying anything by zero always gives zero). So when processing the image, we try to present the important information for analysing the image in white, and irrelevant information, including noise, in black.

The first stage in our system is the *Pre-processing*. This aims to separate in the digital image any information that is not needed for the analysis of the handwriting – dirt, gaps, notes by copyists, margins, holders used to keep the manuscript straight during digital photography, illuminations (see Figure 3), initials – from the information that is crucial for us, that is, the actual text, written in ink.

This first stage actually consists of two smaller steps. The first is the *Binarisation*, which involves converting the colour image into a black-and-white image, in which the background is black (with the binary value zero) and the ink is white (with the binary value one; see Figure 5b). In this case, the input was a manuscript that had been professionally digitised; before the photograph was taken, the manuscript was illuminated with uniform, non-directional light. If the photograph is taken without uniform light, the result will not be as good, as shown in Figure 4b. To achieve the desired effect (Figure 4c), it is necessary to use the more advanced method that we specially developed for our system, which we call "Gaussian binarisation". In simple terms, in order to assess whether a pixel should be white, Gaussian binarisation not only uses sta-

Fig. 3. Information not needed for the handwriting analysis.

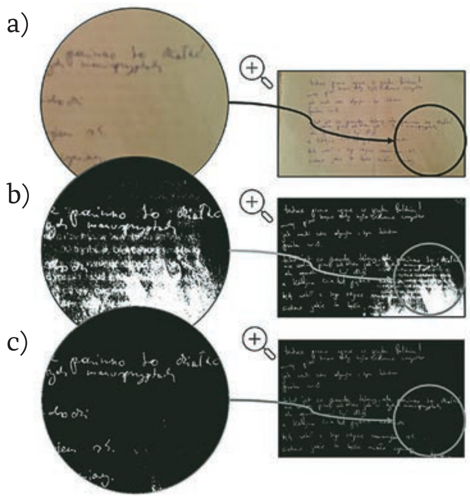


Fig. 4. a) Binarization of a photograph of a manuscript taken with a digital camera without professional lighting; b) Otsu binarization of a); c) Gaussian binarization of a)

tistical data from the whole image, as in Otsu binarisation,[12] but also contextual information; in other words, the value of the pixel depends on the colour of the neighbouring pixels.

a) b)

Fig. 5. a) Manuscript in colour; b) Binarized image.

However, looking at Figure 5b, it is clear that we now have to deal with a new kind of noise: the edges of the sheets of paper have been treated as ink, as has the pagination in the upper right-hand corner, added by someone other than the author of the manuscript. These elements need to be eliminated.

We do this in a second step, which we call *Writing area detection*. Anyone looking at the manuscript, without being able to read it, can easily determine the area covered by the writing and divide it up into different lines. After closer analysis they will be able to group fragments of letters into letters, and letters into words. As mentioned above, to a computer, a picture is only a set of num-

12 S.S. Reddi, S.F. Rudin, H.R. Keshavan, "An optimal multiple threshold scheme for image segmentation", *Systems, Man and Cybernetics, IEEE Transactions*, 1984, SMC-14/4, pp. 661–665.

bers in a certain order; we therefore need to tell the computer to perform all the necessary operations which a human being would perform intuitively, in the form of a series of actions. First, the system must identify groups of pixels within fragments of letters, which we call "connected components", and place them within the smallest possible "bounding box" (see Figure 6).

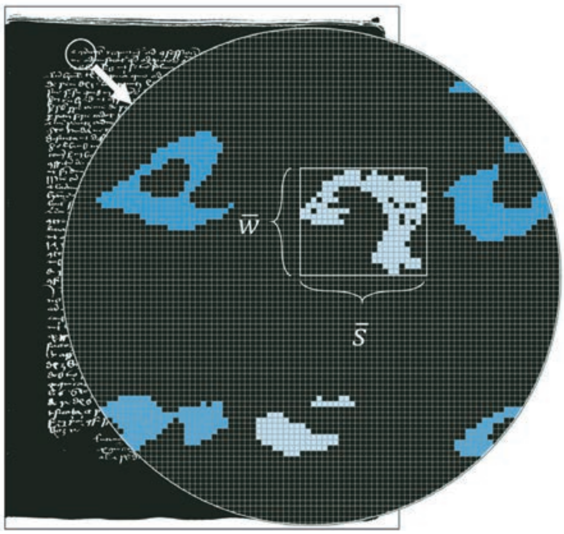


Fig. 6. Detecting connected components and determining their average size.

This makes it possible to determine the approximate dimensions of the letters in the manuscript. Thanks to this procedure, anything that is too small or too big compared to the letters can be rejected at this stage. The algorithm used here is based on the RLE (run-length encoding) method of encoding data series. To illustrate this, we can use the abstract word *aaabbcaaabbbbbbbbbb* as an example. This can be written in pairs of characters consisting of the number of uninterrupted occurrences of the letter, followed by the letter itself, thus: 3*a*,2*b*,1*c*,3*a*,10*b*. As the binary image is a matrix in which each row (or column) consists of a series of zeros and ones, the sequence 110111000101100111, for example, could be written, using the same methodology, as: 2*1*,1*0*,3*1*,3*0*,1*1*,1*0*,2*1*,2*0*,3*1*.

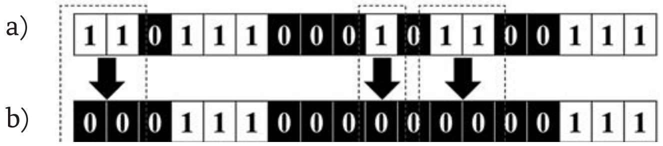


Fig. 7. a) A binary vector; b) The result of replacing a sequence of binary numbers of ones with zeros.

What is the significance of this? In the binary sequence shown in Figure 7a in graphical form, any sequence of binary numbers – for example, those that consist of fewer than three ones – can be replaced by black zeros. The result is Figure 7b. In simple terms, this procedure makes it possible to remove most of the elements that do not form part of the written text under analysis. The operation is performed horizontally and vertically on the input image. The result of this process is shown in Figure 8.

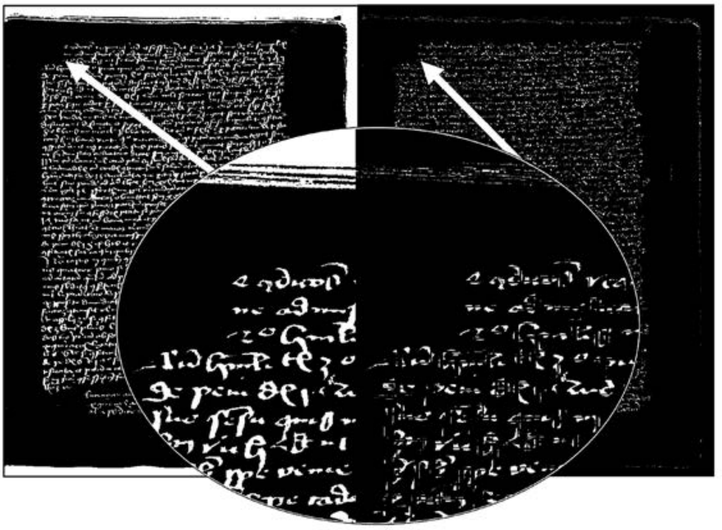


Fig. 8. The binary image before and after replacing a sequence of binary numbers of ones with zeros.

As can be seen, the vast majority of the white area that did not form part of the text has been removed. The text now has bits missing here and there at the corners, but the computer is able to reconstruct these and create a contiguous area containing the text to be analysed, which we call the „binary mask" (see Figure 9b). The input image and the binary mask are the same size, so superimposing one on top of the other gives us the final image, in which all the unnecessary information has been removed and only the parts written in ink which we are interested in remains (see Figure 9c).

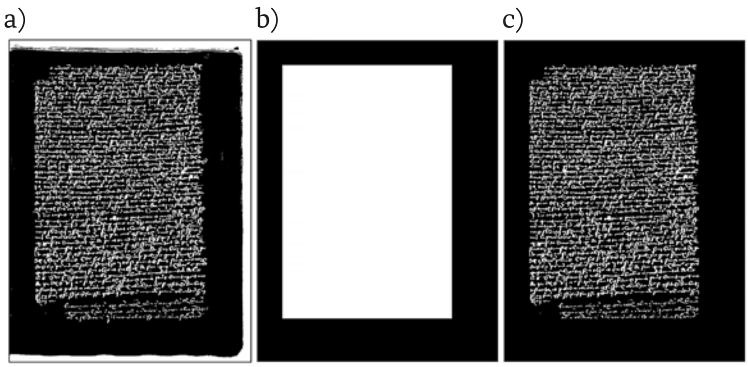


Fig. 9. a) Binary image of the manuscript; b) Detected area of written text; c) Manuscript a) with all unnecessary information removed based on the detected area of written text.

Having been pre-processed in this way, the image can now be used as the input for the second stage in the process, the *Text segmentation*. Like the *Pre-processing* stage, this consists of two steps. The first is *Detecting lines of writing*, the second *Separating the text*. Here, we must touch briefly on the characteristics of Latin manuscripts from the Middle Ages. For a human being, identifying a line of writing is easy, even if the line slopes upwards or downwards; for a computer, on the other hand, this is one of the most difficult parts of the process.

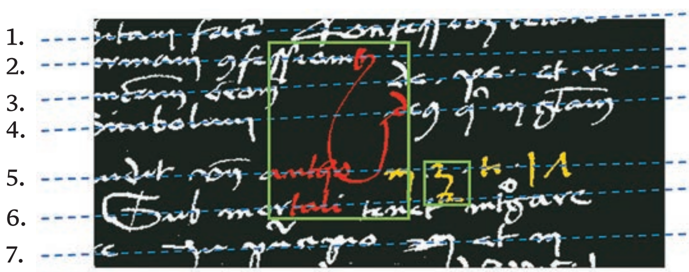


Fig. 10. Example of overlapping letters in a manuscript.

In the Middle Ages codices were a luxury commodity, as were the parchment and ink used to write them. As a result the text was often written very close together, so as to increase the amount of text that could fit on one sheet of parchment. Often parts of the text overlapped (Figure 10). To a computer the part marked red, which covers five lines of text, appears to form a single element; to a human being it is clear that this is due to overlapping letters. Unlike today, the scribe did not rest his hand on the writing material but rather drew the quill across the parchment. As a result the text slopes downwards, forming a curve.

To correctly segment the manuscript into lines of text for further analysis, the computer must detect any areas that are connected to each other and divide them up, assigning them to the correct lines of text. We do this with an algorithm based on an improved Hough transform from 2015.[13] Once again, the words of Thomas à Kempis are fitting here: "the highest does not stand without the lowest." The Hough transform is a modified form of the Rodon transform from 1917, patented by Hough in the United States on December 18, 1962 and used to detect lines in an image. Now widely used in the field of computer vision, this transform is based on the system of polar coordinates successfully used by the Greek astronomer Hipparchus on the island of Rhodes in the second century B.C.[14]

The concept is based on the idea that any line in the Cartesian coordinate system can be written as a point in the polar coordinate system and vice versa, that is, that any point in the polar coordinate system can be written as a line in the Cartesian coordinate system. As we know, any straight line – in other words a linear function – can be expressed by the formula , where a is the slope and b is the y-intercept. If a is greater than zero, the function will be increasing, and if a is less than zero, the function will be decreasing. b is the value at which the function intersects with the Y axis.

13 J.L. Pach, P. Bilski, "A robust binarization and text line detection in historical handwritten documents analysis", *International Journal of Computing*, 2016, 15/3.

14 C.M. Linton, *From Eudoxus to Einstein: A History of Mathematical Astronomy*, New York 2004, p. 52.

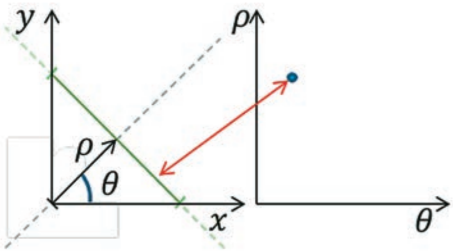


Fig. 11. Transformation of a straight line (green) from the Cartesian coordinate system to the polar coordinate system (blue dot).

"Note the solid green segment of the dotted green line on the left of Figure 11. First, we draw from the origin of the Cartesian coordinate system a line that crosses this solid green segment at right angles (the grey dotted line, one segment of which is black with an arrow on the end). The length of the black segment is the value of the first parameter ρ. The angle which the black line forms with the *X* axis is the second parameter θ.[15]

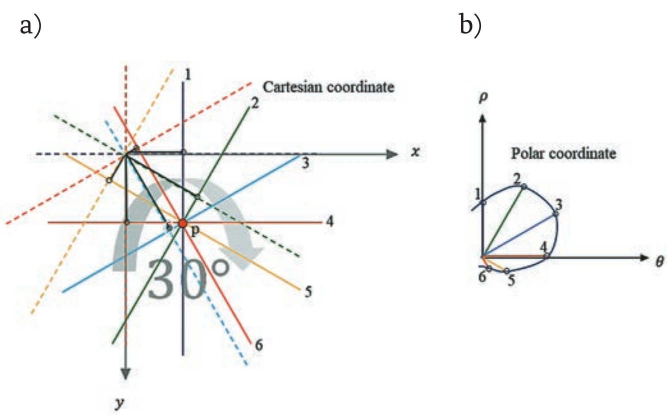


Fig. 12. Rotation of a line around a red point p, and its representation in the polar coordinate system.

The practical implications of this are illustrated in Figure 12. Here, we need to refer to another algebraic correspondence, name-

15 Calculated according to the formula: $x \cdot \cos(\theta) + y \cdot \sin(\theta) = \rho$.

ly that in the Cartesian system an infinite number of lines (such as lines 1, 2, 3, 4, 5 and 6) can be drawn through any point (p, marked in red in Figure 12a). It is also possible to draw a line through this point parallel to the *Y* axis (line 1, purple) and create for it a black segment as we did above (lying along the dotted purple line in Figure 12a) to obtain its length ρ and the angle θ between it and the *X* axis. We then transfer these values in the form of a point to the polar coordinate system. Next, we rotate the purple line clockwise, for example by one degree (or 30 degrees in the case of Figure 12b), to obtain the green line 2. Proceeding in the same manner, marking one point in the polar coordinate system for each line, we obtain the graph shown in Figure 12b. The smaller the rotations, the bigger the number of lines, and so the more points there are in the polar coordinate system. Together these points form a curve.

This is illustrated in Figure 13. On the left-hand side, in the Cartesian coordinate system, we have three points: point 1 is yellow, point 2 brown and point 3 red. On the right-hand side, in the polar coordinate system, these three points correspond to three curves: yellow, brown and red. The illustration shows that the points where the curves intersect on the right correspond to the straight lines on the left. Thus the red and brown curves on the right intersect at a blue point; we already know that points on the right correspond to lines on the left, so on the left we find a blue line on which the red and brown points lie.

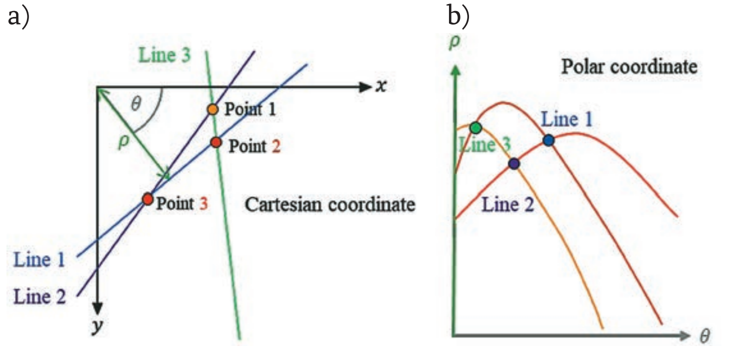


Fig. 13. Conversion of three points in the Cartesian coordinate system to curves in the polar coordinate system.

Thanks to this correspondence it is possible to locate written lines in a text using points in the Cartesian coordinate system. In addition, however, we must split this binary image (from which noise has been removed and which now only contains information about the text itself, written in ink) into horizontal sections. We then select only those points which hypothetically lie in the centre of the line of text in question, as illustrated by the red line *S* in Figure 14.

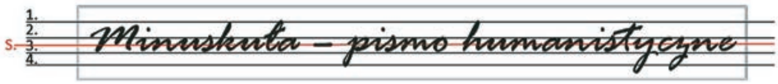


Fig. 14. Minuscule handwriting sample split up using four horizontal lines, with the centre marked.

To obtain these points for the whole text it is necessary to recalculate the sizes of all the objects in the image, as in Figure 6, as the average height and width will have changed significantly after removing the non-text elements. We can divide the remaining text into three groups. The first group contains characters similar to the average size of the elements in the image; this will mainly be fragments of letters. Only this group will be needed for further analysis. The second group contains elements which are significantly larger than the average size; this will include both initials and blocks of connected text, such as the part marked red in Figure 10. At a later stage these will have to be divided up appropriately. The third group contains punctuation marks and diacritics that are significantly smaller than the average size, most often full stops, commas and the like. Figure 15 is a graphic illustration of this division of the text into three groups. The smallest elements will be attached to the letters closest to them in the final part of the *Text segmentation* stage.

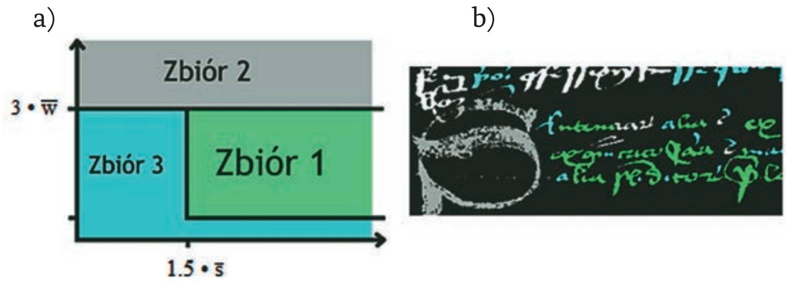


Fig. 15. Graphical illustration of the division of the text into three groups.

Accordingly, we only take green objects (see Figure 15b) into account when looking for lines of writing using the Hough transform. Modern digital representations of manuscripts are usually very high resolution so the fragments of letters often consist of a huge number of pixels, as can be seen in Figure 6. The line *S* in Figure 14, which is what we are looking for, essentially cuts through the middle of them. So in order to avoid unnecessary calculations by the computer, which might introduce false information, the green objects are divided into blocks and one central point is determined for each block, as shown in Figure 16.

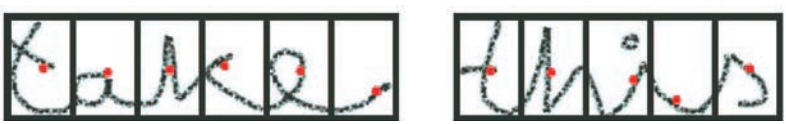


Fig. 16. The points used in the procedure for identifying lines of writing are marked red.

In simple terms, if enough points lie on the same straight line, the Hough transform will identify it; if not, an additional algorithm is needed to highlight the information for the Hough transform. This solution is not ideal but thanks to this step it is possible to identify a potential set of lines of writing for the entire manuscript. Often this set is incomplete, so another step is needed to reconstruct and eliminate any redundant lines of writing. Figure 17 shows the final result of the *Detecting lines of writing* step; red boxes are used to show where different letters overlap. In order to divide the text into lines of writing we need to separate these overlapping letters and assign them to the correct line.

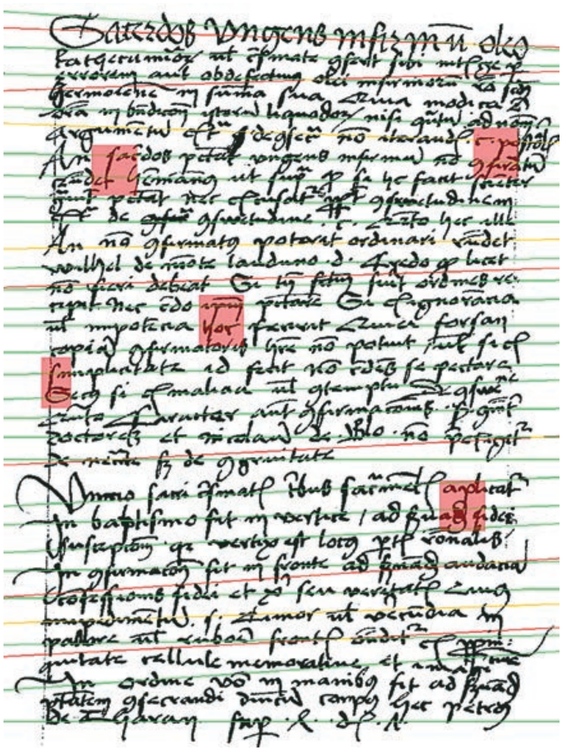


Fig. 17. Lines identified by the traditional method are marked red, those marked by our improved version are marked green. Lines identified by both methods are marked yellow. Red boxes show where letters overlap across more than one line of writing.

Thanks to the fact that we have a complete set of the lines of writing, we can now move on to the second step in the *Text segmentation* stage, namely *Separating the text*. Having located all the lines of writing, the computer checks whether there are any fragments of text linked to each other which pass through at least two lines of writing. These fragments must be divided up, which requires a process known as "skeletonisation". Skeletonisation involves reducing the image by making successive "passes". This is a process akin to sanding something down on each side, step by step, until you get the thinnest object possible, such that if you sanded it down any more it would disintegrate. This is illustrated graphically in Figure 18.

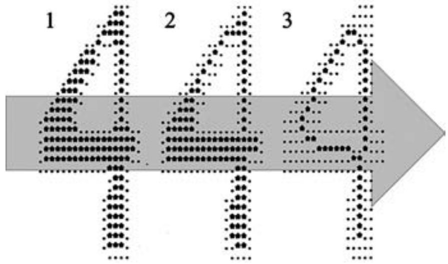


Fig. 18. Three passes of skeletonization.

After obtaining these skeletonised images of fragments of the text, the points of intersection are removed so that the fragments can be assigned to the nearest lines of writing that intersect them. These points of intersection are indicated with red circles in Figure 19c.

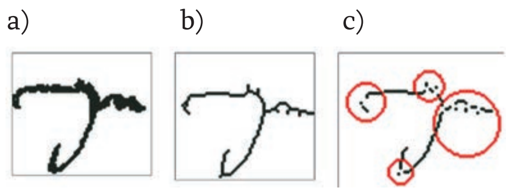


Fig. 19. a) Fragment of text; b) Skeletonised image of fragment a); c) Removal of intersection points in b)

The remaining parts (pixels) that were removed during skeletonization, and the intersection points, are then replaced and attached to fragments that have already been assigned to lines of writing, as in Figure 20. This process is commonly known as "colouring".

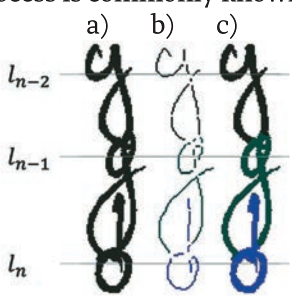


Fig. 20. Colouring a fragment of text: a) Original fragment;
b) Skeletonised image of fragment with intersections removed;
c) The fragment coloured, i.e. assigned to the correct line of writing.

Figure 21 gives an example of the result of the first two stages, the *Pre-processing* and the *Text segmentation*.

Fig. 21. Graphical illustration of the detection and colouring of lines of writing by the system.

A human being looking at the manuscript would intuitively divide the lines of writing up further into sentences, then words, then letters made up of strokes. However, here it is not necessary. In our system it is not important *what* is written but rather *how* it is written. The basic unit of writing still big enough to allow efficient recognition of handwriting samples is the individual line. The father of modern linguistics, Ferdinand de Saussure, born at the beginning of the second half of the nineteenth century, already had a strong suspicion about how the human brain functioned, long before the first computer was built. His definition of a sign is as follows:

> "The linguistic sign unites, not a thing and a name, but a concept and a sound-image. The latter is not the material sound, a purely physical thing, but the psychological imprint of the sound, the impression that it makes on our senses."[16]

De Saussure noted that the human brain must contain a template for each sign which people use intuitively and if necessary can reproduce – a kind of memory cell in which the set of distinctive features necessary for identifying a sign are stored. As a result the brain can tell whether the written sign matches the template or not. For example, in theory it is not difficult for someone who knows the Latin alphabet to identify a capital letter *A*. But when that person writes the letter *A* themselves, it will not only contain these distinctive features but also certain features characteristic of the writer. The characteristic features of their handwriting may include the use of italics, the thickness of the writing, how round certain letters are – features that are characteristic for them and them alone, features that are unique, a kind of marker.

The same applies to all the skills that an individual has mastered in the course of their life: they will have learned and remembered those skills in a certain way. Everything that they do, they do according to a template which is limited by their own physical condition – from how they write (how often they use certain words, their syntax, their use of anachronisms) to how they walk or the body language they use. On this basis an intelligent computer system is able to identify a person from a video recording of them[17] or a handwriting sample – by how they draw the upward stroke of the letter *b*¸ for

16 F. de Saussure, *Course in general linguistics;* transl. Wade Baskin, New York 2011, p. 66.

17 Y. Pratheepan, P.H.S. Torr, J. V Condell, G. Prasad, *Body Language Based Individual Identification in Video Using Gait and Actions*, Berlin, Heidelberg, 2008.

example. Human beings often subconsciously try to reconstruct the patterns stored in their brain. This is the basis for how intelligent computer systems work: they are able to capture these patterns. Indeed, those studying the psychology of handwriting go a step further and claim to be able to identify the health, emotional state or age of the writer on the basis of their handwriting.

To sum up, after the initial image processing and removal of everything that is unnecessary for further analysis of the handwriting, the text is segmented up into lines of writing. This allows us to move on to the next stage in the system, the *Extraction of handwriting features.* This stage is again divided into two steps. The first step is *Using the scale-invariant feature transform (SIFT),* in which all the individual features of a particular writer's handwriting are extracted on the basis of a sample. The second is the construction of a dictionary containing digital representations of all the samples of handwriting that have been analysed.

The third step involves the technique known as "template matching", which includes a set of algorithms that form the basis for the complex transform SIFT from the field of digital image processing. This itself is a specialised piece of software/programme for identifying characteristic features or "keypoints" in images. SIFT is used in a wide range of applications besides the analysis of handwritten and printed texts, such as object recognition and tracking, building maps, robot localisation, creating panoramic images, reconstructing three-dimensional objects, identifying human gestures and adjusting images in films.

To explain how traditional template matching works, we first need to define two types of image: "templates" and "scenes" (see Figure 22). The task is to identify where a template occurs on a scene. If a template occurs, the system should determine its position; if it does not, the system should send an appropriate message to the user.

Fig. 22. a) Digital image of a Latin manuscript,[18] forming the scene; b) Digital image of an initial and magnified version of the image; c) The detected template on the scene.

Figure 23 illustrates how the computer finds a template on a scene, showing the representation of the image and its scale. The image has been enlarged three times (1-3) so that we can see the beginning of the binarised area highlighted in the red box.

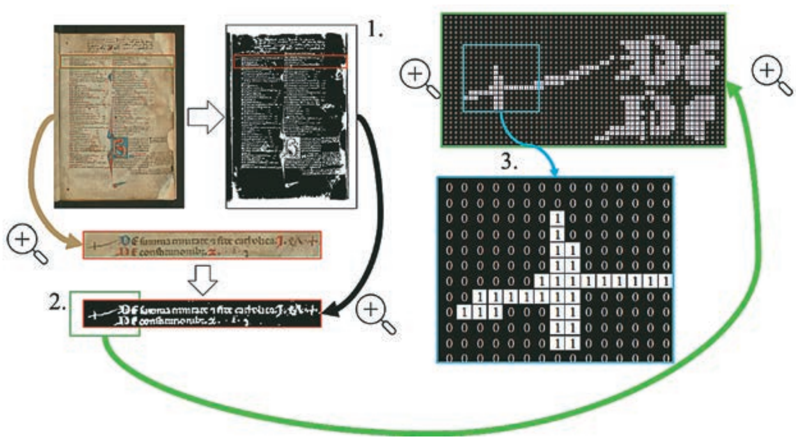


Fig. 23. Binarisation of a Latin manuscript and graphic representation of the binary matrix of part of it.

To illustrate the size of the digital image of the Latin manuscript:

---

18 BJ MS 342 III.

the image is 637x898 pixels, and it represents just ten per cent of the original image taken from the holdings of the National Library of Poland. That gives us a total of more than half a million pixels. The template we are looking for is 90x112 pixels, which is more than 10,000 pixels. In terms of scale, this is like looking for tyre tracks on a football field. The computer applies the template to the top left-hand corner of the image, touching the upper left-hand edges of the scene, and then calculates the cross-correlation between the template and the overlapping part of the scene. The cross-correlation value is the percentage of the images that overlap. The computer saves this value, then moves the template one pixel to the right and repeats the calculation. When it reaches the right-hand edge of the image, the template returns to the left-hand edge, one pixel further down. It repeats this until the algorithm calculates the set of correlation values for all matches of the template to the scene; this is known as the „correlation matrix". The algorithm then searches through the correlation matrix and if it finds a value close to 100 per cent, then it has determined the exact coordinates of the object on the scene.

Clearly this process involves a huge number of calculations. Moreover, the method is somewhat flawed: the algorithm will not work if the template is rotated by even a few degrees or if it is of a different scale to the object on the scene. In 1999, to overcome these limitations David Lowe developed the complex SIFT algorithm,[19] a scalable transform of the features of an image.

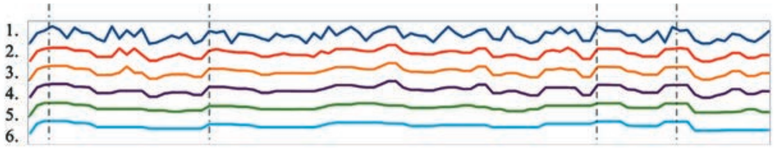
The SIFT algorithm works on the same principle as human sight. Image processing is analogous to the perception of stimuli by a human being suffering from short-sightedness but wearing corrective glasses, thanks to which they can see clearly. If they spend enough time in the company of a second person, their brain will build a detailed template pattern of that person on the basis of their appearance, their body proportions, height, the shape of

19 D.G. Lowe, "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, 2004, 60/2, pp. 91-110.

their limbs and other distinguishing features (for the sake of simplicity we refer only to visual stimuli). If the short-sighted person – this time without their glasses – meets the other person again it is highly likely that they will still be able to pick them out in a crowd. This is because the above-mentioned features will still be visible to them, even if the image is unfocused; the proportions of the person's body will not change even if the person turns or moves closer or further away from the viewer.

Figure 24 illustrates this. With the right glasses, the short-sighted person will see a sharp image, rich in detail, as represented by the blue line 1. If the glasses are too weak the person will see less clearly, as represented by the orange line 2. The remaining lines show the results for increasingly weak glasses, right down to no glasses at all. However, it is worth noting that even without glasses some of the individual peaks (specific features), marked with vertical dotted lines, are still visible.

Fig. 24. A blue line smoothed by median filtering (2-6).



SIFT comprises five stages. The first stage is generating auxiliary images – known as the pyramid – which are needed to enable independence from scale and to transform them into Differences of Gaussians (DoGs). The second stage is detecting points of interest (local extremes) in these images as a set of potential keypoints. The third stage is selecting keypoints, and the fourth finding the value of the following parameters for the keypoints: location coordinates, orientation and scale. The fifth and final stage is computing a descriptor vector for each keypoint.

The first stage is the pyramid. This consists of the original image, followed by a copy of the image half its size, followed by another copy half the size of that, and so on (see Figure 25). Once a pyramid has been constructed, several copies of it are made and then „blurred": each copy has a higher blurring coefficient in order to hide certain details and bring others into prominence, depending on the scale.

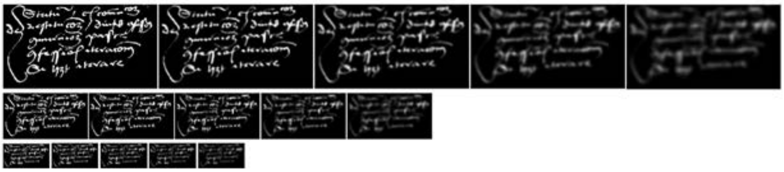


Fig. 25. Image reduction and blurring by using an increasing blurring coefficient.

This simulates the example that we gave above of the short-sighted person. Having created copies of the images in different sizes and with different degrees of focus, these images are now used to create DoGs (see Figure 26). This involves creating a new image using the differences in values between two images of the same size but with different levels of blurring. In this newly created image there will be peculiarities (local extremes). We then need to search the DoGs for potential keypoints. The clearest peculiarities belong in the initial set of keypoints, that is, the characteristic features of the input image.

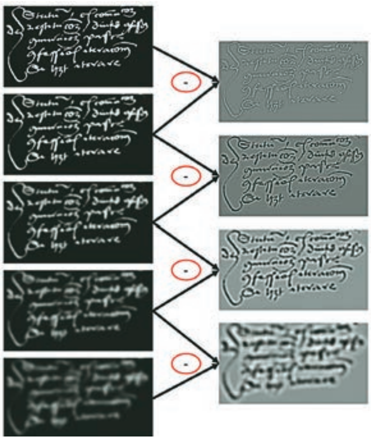


Fig. 26. Differences of Gaussians (DoGs) for one dimension.

A process of selection follows, removing candidates where the contrast is too low or which are located on the edges of the image being analysed. This gives us the final set of keypoints, which can be shown as vectors – the red arrows in Figure 27. Each arrow has a specific length (scale), direction (orientation) and specific coordinates.

Fig. 27. a) Coordinates of keypoints; (b) Direction and length of keypoints.

While the coordinates are important when looking for the distribution of keypoints relative to each other in the image, they are unnecessary when it comes to recording features of handwriting. This is because recording features of handwriting is a kind of statistical operation concerning the distribution of the directions and sizes of keypoints. These are recorded in the form of two "frequency of occurrence" charts for a specific orientation and scale. An example for a single handwriting sample is in Figure 28.

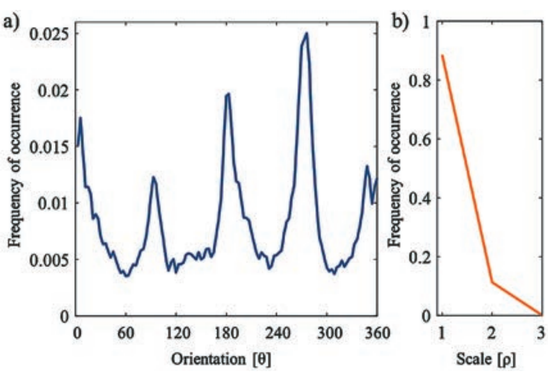


Fig. 28. Graphic representation of a single handwriting sample in the form of two charts: (a) orientation; (b) scale.

If a digitised handwriting sample is already available, we can now move on to the second step in the *Extraction of handwriting features* stage, namely *Constructing a dictionary*. Each individual handwriting sample is saved as a single vector, containing the scale and orientation values. Next, we create a database of all the handwriting samples to which the system has access. Let us take, for example, the digital representation of the features of the handwriting samples from the *Chronicle of Father Stefan Ranatowicz*. Figure 29 shows the first 200 parchment sheets[20].

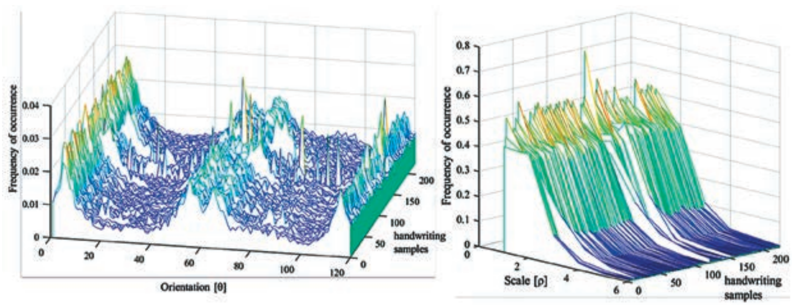


Fig. 29. Graph showing handwriting samples from 200 parchment sheets from the Chronicle of Father Stefan Ranatowicz[21].

The irregularities, which are visible to the naked eye, clearly relate to handwriting samples from the other writers, namely Father Michał Aquilin Gorczyński and Father Krzysztof Piasecki.

The final stage in the system is the *Classification*. This is divided into three steps: *Dividing up the data, Calculating similarities* and *Making the decision*. To verify the accuracy of this final stage we need to divide up the handwriting samples in the dictionary that we have created into a "training dataset" and a "validating dataset", each of the appropriate size.

In this paper we divide up the data using a process of cross-validation. This involves dividing up the data piece by piece into training and validating datasets. In one sample, the label indicating

20 S. Ranatowicz, Casimiriae civitatis, MS BJ 3742 III.

21 K. Łatak, M. Pręgier, J. Pach *Z problematyki kodykologicznej i paleograficznej…*, op. cit.

the author is hidden. The computer is then instructed to find the sample which is most similar to it among all the other samples. The computer finds a possible candidate and says what it thinks its label is. If the computer gives the correct label, the identification is considered correct; if not, it is considered incorrect. The same is then done for all the other samples of writing in the dictionary, one by one. The accuracy of the classifier is the arithmetic mean of all the decisions it takes. This method allows us to determine the effectiveness of the classifier regardless of the specific dataset, and also makes the results objective.

Figure 30 presents seven handwriting samples (marked light grey). In the first step, the label of the first sample is hidden. As a result, the sample goes in the validation set (marked orange) and the rest of the samples go in the training dataset (marked dark grey). In steps 2 to 7, the next samples are treated the same way, until all the handwriting samples have been used up.

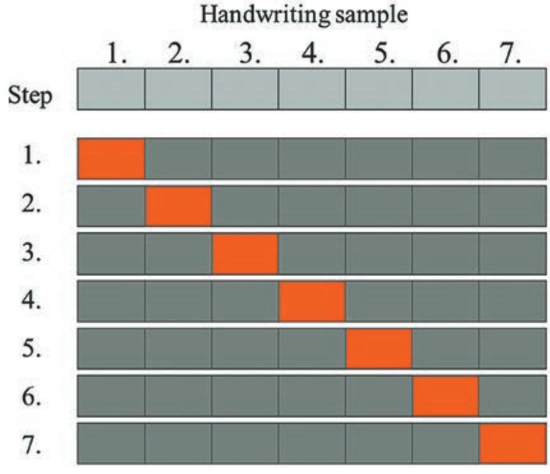


Fig. 30. Illustration of cross-validation.

The other two steps in this stage are calculating similarities and making the decision. We will illustrate these two steps together, without going into the detail of the mathematical approach used.

In the Cartesian coordinate system it is not difficult to imagine the distance between two points (see Figure 31a). The "nearest neighbour search" (NNS) method involves calculating the shortest Euclidean distance between the samples analysed. Usually, finding one "neighbour" is enough to identify the correct label, but this is not always the case. Sometimes we need to take more than one into account in order to obtain a more reliable result. Figure 31b gives an example, where the grey arrows with dotted lines are also taken into consideration. Of course it is possible that the shortest distance is greater than the limit value or "threshold". In this case, where the computer finds a sample with too little similarity it refrains from giving a label to the nearest neighbour or declares that the sample is unknown and that there are no more handwriting samples from the same writer in the dictionary.

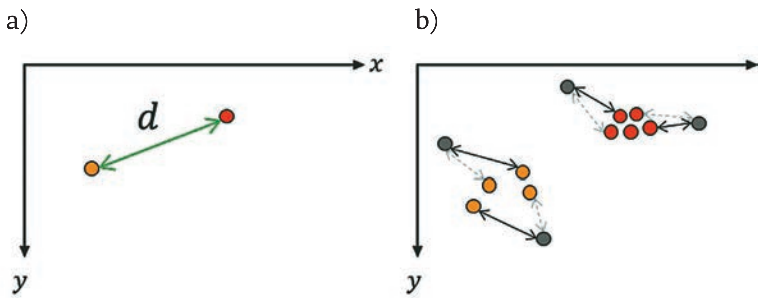


Fig. 31. a) Distance d between two points; b) Illustration of the shortest distances between samples.

Two databases were used to verify the effectiveness of our system. The first was created from five codices dating from the twelfth to the fifteenth centuries, available digitally from the website of the National Digital Library in Warsaw.[22]

- MS 12511 II – created [ca. 1175-1200]
- MS 3307 II – created [ca. 1160]
- MS BOZ 36 – created [ca. 1245-1255]

22 MS 12511 II, MS 3307 II, MS BOZ 36, 1197 V MS.

- MS 1197 V - created [1415]
- MS 3469 II - created [ca. 1401–ca. 1500]

In the base in question there are statistically 15 manuscripts in a code, and most of them have two columns, which gives 98 columns of text. Yet, it should be borne in mind that these columns were narrow. The division of all documents into lines of writing yielded 3510 images that were the input data to extract the handwriting and to examine the database of mediaeval texts. In the database of Latin manuscripts, for the classifier of the nearest neighbour the threshold of rejection of authorship is 0.25. The efficacy of the classifier for data prepared in this way was 99.66%.[23] The classifier made only one mistake when it misattributed one of the documents. The effect of such misattribution can be seen in Fig 32.

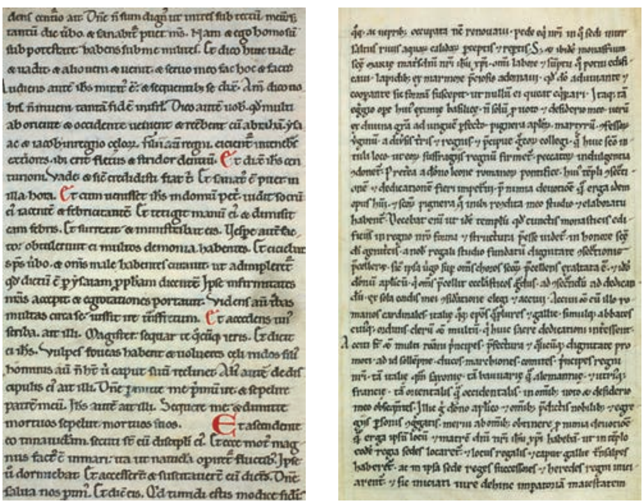


Fig. 32. Two different samples of handwriting attributed by the system to one person

The second was the database of contemporary manuscripts in English of the Computer Science Institute at the Modified database of the Institute of Informatics and Applied Mathematics (MIAM) at the University of Bern. It contains contemporary texts in English

23 J. L. Pach, *Identyfikacja autora rękopisu łacińskiego…*, op. cit.

handwritten by 31 persons, nine or ten pages each, that is 2520 lines of handwriting. For this base the efficacy of the classifier equalled 98.8%. As before, the results of misattribution can be seen in Fig. 33.

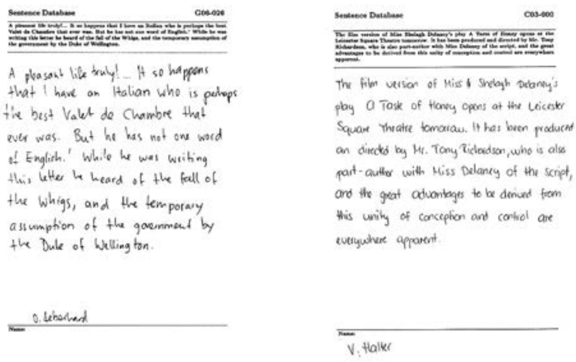


Fig. 33. Text written by two different persons misattributed to one person by the classifier. The manuscript on the left was wrongly attributed to the writer on the right.

In order to show the capabilities of the system these results were compared with those from other systems using the same MIAM database. To analyse the MIAM database a classifier of *k* nearest neighbours with the Soft TOP-N decision strategy. This strategy consists in indicating *N* documents from the dictionary that are closest to the analysed sheet. This decision is seen as correct if at least one of the documents of *N* closest ones was created by the author of the sheet in question. This strategy differs from the nearest neighbour method because it does not reject the authorship hypothesis when the similarity is too small, and it only indicates the most similar manuscript. For Soft Top-1 the results will be identical with the weighted nearest neighbour classifier. Tab. 1 shows the results of the MIAM database classification using a number of algorithms used by other researchers. As can be seen, the approach proposed proved to be the most accurate of all approaches investigated so far. Therefore it can be used to analyse contemporary handwriting.

TAB. 1 RESULTS OF CLASSIFICATION BASED ON MIAM.

Approach	Soft-Top-1
SOH[24]	78.4
Grapheme emission(GE)[25]	80
Countour-hidge(GH)[26]	81
GH+GE[27]	88
GMF[123][28]	90
Siddiqi[124][29]	91
Line fragment[30]	93.7
Countour-hidge(GH)[31]	94
SDS[32]	94.2
LPQ[33]	96.7
Quill_Hindge[34]	97
SDS+SOH[35]	98.5
Presented in the article[36]	98.8

---

24 X. Wu, Y. Tang, W. Bu, "Offline text-independent writer identification based on scale invariant feature transform", *IEEE Transactions on Information Forensics and Security*, 2014, 9/3, p. 526-536.

25 M. Bulacu, L. Schomaker, "Text-independent writer identification and verification using textural and allographic features", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, 29/4, p. 701–717.

26 Ibidem.

27 Ibidem.

28 X. Li, X. Ding, "Writer identification of chinese handwriting using grid microstructure feature", *Advances in Biometrics*, 2009, p. 1230–1239.

29 I. Siddiqi, N. Vincent, "Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features", *Pattern Recognition*, 2010, 43/11, p. 3853–3865.

30 G. Ghiasi, R. Safabakhsh, "Offline text-independent writer identification using codebook and efficient code extraction methods", *Image and Vision Computing*, 2013, 31/5, p. 379–391.

31 A.A. Brink, J. Smit, M.L. Bulacu, L.R.B. Schomaker, "Writer identification using directional ink-trace width measurements", *Pattern Recognition*, 2012, 45/1, p. 162–171.

32 X. Wu, Y. Tang, W. Bu, "Offline text-independent writer identification based on scale invariant feature transform", op. cit.

33 D. Bertolini, L.S. Oliveira, E. Justino, R. Sabourin, "Texture-based descriptors for writer identification and verification", *Expert Systems with Applications*, 2013, 40/6, p. 2069–2080.

34 A. A. Brink, J. Smit, M. L. Bulacu, L. R. B. Schomaker, "Writer identification using directional ink-trace width measurements", op. cit.

35 X. Wu, Y. Tang, W. Bu, "Offline text-independent writer identification based on scale invariant feature transform," op. cit.

36 J.L. Pach, *Identyfikacja autora rękopisu łacińskiego*, op. cit.

The influence of the size of the learning dataset on the efficacy of the writer identification was also examined for Latin manuscripts. For such a base of old texts the system remains stable provided the dataset size is limited to 10 per cent – otherwise the efficacy is reduced to 97 per cent, and then drops down drastically. The size of the learning dataset is the percentage of all lines of writing of a given writer.

Translated by Nick Ukiah

SUMMARY

This article presents the architecture of a system for automatically identifying samples of handwriting. The system allows for the fact that acquiring images is particularly difficult from Latin manuscripts due to the specific nature of such texts, and successfully extracts only such information as is necessary to analyse the features of the handwriting in the sample. The system proposed here is not only able to recognise correctly samples of a single author's handwriting in different scales, but also, after appropriate modification, to identify the features of the handwriting in which the text was written. This can also help with an initial assessment of the date when the text in question was created.

KEYWORDS: handwriting identification, Hough transform, scale invariant feature transform, pattern recognition, projection profiles