Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Image-based historical manuscript dating using contour and stroke fragments

Sheng He [a,*], Petros Samara [b], Jan Burgers [c], Lambert Schomaker [a]

[a] Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen, PO Box 407, 9700 AK Groningen, The Netherlands
[b] Department of History, University of Amsterdam, Spuistraat 134, 1012 VB Amsterdam, The Netherlands
[c] Huygens Instituut voor Nederlandse geschiedenis, PO Box 90754, 2509 LT, The Hague, The Netherlands

ARTICLE INFO

Article history:
Received 23 November 2015
Received in revised form
2 March 2016
Accepted 25 March 2016

Keywords:
Historical manuscript dating
Writer identification
Contour fragment
Stroke fragment
Handwriting style

ABSTRACT

Historical manuscript dating has always been an important challenge for historians but since countless manuscripts have become digitally available recently, the pattern recognition community has started addressing the dating problem as well. In this paper, we present a family of local contour fragments (*k*CF) and stroke fragments (*k*SF) features and study their application to historical document dating. *k*CF are formed by a number of *k* primary contour fragments segmented from the connected component contours of handwritten texts and *k*SF are formed by a segment of length *k* of a stroke fragment graph. The *k*CF and *k*SF are described by scale and rotation invariant descriptors and encoded into trained codebooks inspired by classical bag of words model. We evaluate our methods on the Medieval Paleographical Scale (MPS) data set and perform dating by writer identification and classification. As far as dating by writer identification is concerned, we arrive at the conclusion that features which perform well for writer identification are not necessarily suitable for historical document dating. Experimental results of dating by classification demonstrate that a combination of *k*CF and *k*SF achieves optimal results, with a mean absolute error of 14.9 years when excluding writer duplicates in training and 7.9 years when including writer duplicates in training.

1. Introduction

Handwritten historical documents are the most important sources of information about the past, especially where the more distant past is concerned, before the wide spread dissemination of printing and semi-mechanical text production. Increasing numbers of such documents are currently being digitized and stored in the computer, as in the Monk system [1], which contains more than 100K scanned page images. Thanks to this development, pattern recognition techniques can now be applied to solve historical document problems, which has already been attempted at length in the case of writer identification [2–5], word spotting [6,7] and character recognition [8,9]. These methods aim to provide efficient tools for scholars in the humanities to discover informative patterns in large digital collections. The Monk system [1], providing a web-based search engine for characters and words annotation, recognition and retrieval, can serve as an example.

Historical manuscripts loose much of their usability as sources if they cannot be dated with some accuracy. However, the fact is that most of them, especially those from the Middle Ages, do not carry any explicit date information. Often the only way to date these manuscripts is by inferring the year or period of origin from the characteristics of the handwriting they contain. Traditionally, this type of historical document dating has been the prerogative of paleographical specialists, basing themselves on years of experience and the non-verbal intuition acquired from it, rather than on objective criteria. Manual script dating is not efficient, as paleographical expertise is comparatively rare and, moreover, it is no exception for experts to arrive at conflicting conclusions when dating the same manuscript. Therefore, automatic script dating offers great promise for countless scholars working with undated handwritten historical sources.

The main motivation of using the computer to date historical manuscripts is to exploit patterns of handwritten texts that correlate with temporal information. This problem is similar to the "visual dating" problem in computer vision, such as historical color image dating [10], estimating the date of historical cars [11] and human age estimation based on face images [12,13]. The aim of visual dating is to mine the visual patterns that are specific for a

* Corresponding author. Tel.: +31 50 363 7410.
E-mail addresses: heshengxgd@gmail.com (S. He), petros.samara@huygens.knaw.nl (P. Samara), Jan.burgers@huygens.knaw.nl (J. Burgers), L.Schomaker@ai.rug.nl (L. Schomaker).

http://dx.doi.org/10.1016/j.patcog.2016.03.032
0031-3203/

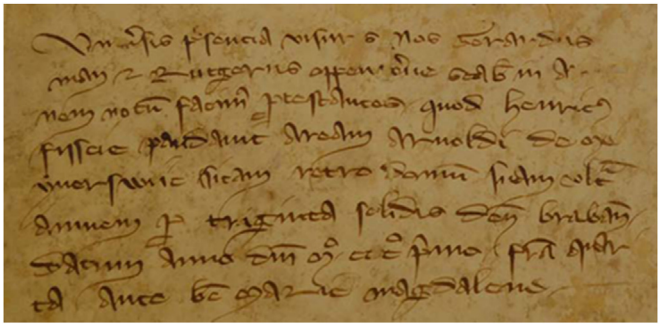

Fig. 1. An example of a charter in the MPS data set.

certain period in time [10] and to track and trace visual styles that change gradually over time [11].

We have proposed a number of features [2,14–16] to capture handwriting styles. However, there is one aspect of the visual appearance of handwritten samples that has not been addressed yet. In Fig. 1, a sample is shown. As we can see, the visual appearance is dominated by long curved stroke elements crossing other ink stroke traces in an irregular manner. Such a complicated thread structure was not covered by the junction feature [16,17] nor by other methods [2,14,15]. In addition, the existing methods concern low-level features, which cannot capture the properties of mid-level graphemes or stroke information. The research questions then are as follows: (1) How to define a feature that addresses the aspect of style at intermediate scale? (2) Which type of properties of handwritten strokes in historical documents contain the temporal information that can be used for dating? (3) What degree of feature complexity is required to obtain the optimal year estimation performance?

In this paper, we propose a family of local contour and stroke features and their application to historical document image dating. These features are small fragments of contours and strokes, called k contour fragments (kCF) and k stroke fragments (kSF), respectively. The fragments in kCF are the contour fragments resulting from a combination of a number of k consecutive primary fragments generated by the discrete contour evolution (DCE) [18] and the fragments in kSF form a segment of length k of a stroke fragment graph (SFG). The larger the number k of contour and stroke fragments in kCF and kSF, the more complex the contour and stroke fragment structures it can capture. We use the relative coordinates of the fragment points of kCF as the feature vector and use the polar stroke descriptor (PSD) proposed in [17] to describe the kSF.

The proposed kCF and kSF can be considered as grapheme-based representations and have several attractive properties: (1) kCF and kSF cover short contour and stroke fragments of the connected components in handwritten documents, which are probably shared between different characters and allographes. The statistical distribution of these small fragments can capture the handwriting style of historical documents. (2) For a certain range of k, both kCF and kSF can discover the meaningful and intermediate complexity patterns in a large connected component which may span several lines due to touching ascenders and descenders in cursive handwriting. (3) The descriptors of the kCF and kSF are insensitive to the scale and rotation of document images, which are very important properties in historical document analysis because historical documents are often digitized with different resolutions and font sizes in different documents are also different, making them sensitive to scale and rotation.

Inspired by the bag-of-words model [19], we construct codebooks of kCF and kSF with different complexity degrees k, each of which capture statistical information with different degrees of complexity of local fragments. All the kCF and kSF detected from handwritten images are mapped into the trained corresponding codebooks to form statistical histograms, the normalizations of which are the final representations of handwritten documents. We demonstrate the flexibility and power of kCF and kSF by applying them to historical document dating using the MPS data set [20].

We organize the rest of the paper as follows. Section 2 provides a review of related work on features used in writer identification and historical document dating. We introduce our MPS data set in Section 3. The details of the proposed kCF and kSF are outlined in Section 4 and Section 5, respectively. We evaluate the kCF and kSF on the MPS data set in Section 6. Finally, we conclude this paper in Section 7.

2. Related work

Various features have been proposed for handwritten document analysis in the previous studies. In this section, we first provide a brief review of the features used for writer identification. Previous studies on historical document dating are summarized in the second part.

2.1. Features used in writer identification

Features used in writer identification can be typically divided into two groups: textural-based and grapheme-based features. Textural-based features extract the texture, curvature or slant information from the entire document image, while grapheme-based features are the normalized histograms of individual graphemes based on trained codebooks, following the bag-of-words framework.

2.1.1. Textural-based features

Several types of textural-based features have been proposed in the literature, which can be roughly categorized into contour-based texture methods and filter-based texture methods.

The Hinge kernel on edges of the text can reflect the writing style [21] and the corresponding Hinge feature which is a distribution of the Hinge kernel on the entire document image has been used for writer identification in [14,15]. The Hinge feature has been extended to Δ^nHinge [22] to achieve the rotation-invariant property. In order to capture the width of ink traces, the Quill feature has been proposed in [2], which is a probability distribution of the relation between the ink direction and the ink width.

Spatial filtering techniques have been used to extract texture features from a handwritten text block. In [23], the Gabor filters and gray-scale co-occurrence matrices have been applied to writer identification. XGabor filters [24] which are obtained by modulating a centered sinusoid with a Gaussian have been used in Persian language writer identification. The oriented Basic Image Features (oBIFs) at two scales have been proposed in [25], using a bank of six Derivative-of-Gaussian filters.

2.1.2. Grapheme-based features

The COnnected-COmponent COntours (CO^3) has been proposed in [14] and applied to isolated uppercase handwritten documents with clear character segmentation. This was extended to lowercase handwriting in [15] by splitting cursive handwriting at the minima in the lower contour that are proximal to the upper contour, called Fraglets. Redundant small patterns of handwritten text were proposed in [26]. Recently, synthetic graphemes based on the beta-elliptic model were used for Arabic writer identification [27]. Singular structural regions in handwriting texts, such as junction regions, were extracted and a junction feature was

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

proposed in [16] to capture the information in junction regions for cross-script writer identification between Chinese and English.

Contour fragments are discriminative visual parts and are used in many applications. The triple adjacent contour segments of handwritten texts was used for language identification in [28]. In [29], contour fragments with a fixed length were extracted from contours of handwritten texts and used for writer identification. In [30], a new shape representation called bag of contour fragments (BCF) was proposed by describing shape contour fragments using shape context [31] and encoding each contour fragment into a shape fragment codebook for shape recognition. The contours of leaf shapes have also been used for plant identification [32].

2.2. Historical document dating

The historical document dating problem has been studied recently in [20,33–35]. Our previous work in [20] used a combined global and local regression method based on the Hinge and Fraglets features to estimate the year of origin of historical documents from the MPS data set. A similar method was proposed in [33] based on the "Svenskt diplomatariums huvudkartotek" collection, consisting of scanned images of charters from the medieval period kept in the Swedish national archive (but not necessary produced in Sweden). A method to date Syriac documents was proposed in [34], using inkball models on a collection of securely dated letter samples from the period between 500 and 1100 CE. In [35] a method to infer the date of printed historical documents from their scanned page images was developed, using convolutional neural networks (CNN) on a data set from the Google books corpus [36].

3. Medieval Paleographical Scale (MPS) data set

The Medieval Paleographical Scale (MPS) data set was first introduced in [20] for historical document dating and the evolution of writing styles within this data set was studied from a paleographical point of view in [37]. The MPS data set consists of images of charters produced between 1300 and 1550 CE in four cities in the Low Countries: Arnhem, Leiden, Leuven and Groningen. Geographically, these four cities can be regarded as a cross section of the Medieval Dutch language area, and the development of writing styles visible within this data set therefore as approximating the development of writing within this area in general. Fig. 1 shows an example of a charter (from Arnhem) in the MPS data set.

As the evolution of writing is a rather slow process, not every year in the period under consideration (1300–1550 CE) needed to be taken into account. The charters were therefore collected according to a sampling interval method. "key years" were set at every quarter century such as 1300, 1325, 1350,…, 1550. Only explicitly dated charters produced in these key years and within a period of five years before or after them that were determined to have been written in one of the four cities mentioned before were included. There are currently 2858 charter images in the MPS data set, grouped around 11 key years. Table 1 shows the numbers of documents over the key years and the four cities. The frequencies are the natural counts of appearance in archives, which have an underlying (historical) cause.

There is a clear general trend discernable in the development of writing styles. Fig. 2 shows four characters ("a", "d", "g", and "p") written in consecutive key years. The handwriting style of these characters shows a clearly datable evolution, for example, double "a" being replaced by single "a" from 1375 onwards. The charters were mostly written by professional scribes, whose working careers could cover several decades. Each writer has an individual writing style, resulting in a distinct average writing style for each key year. There is, nevertheless, also, a general trend in the development of writing styles – the evolution of writing styles being a gradual process. The writing styles found in nearby key years is always more alike than in key years further removed from each other.

4. *k* contour fragments (*k*CF)

The contours of handwritten texts encapsulate the handwriting style and a wide variety of approaches have been proposed to extract features on writing contours, such as the CO^3 [14], chain codes [26] and contour fragments [29]. In this section, we propose a novel framework to extract contour fragments, called *k* contour fragments (*k*CF for short), on contours of handwritten texts in historical document images. Our method is more flexible and insensitive to scale and rotation transform. The computational procedure will be presented in the following sections.

4.1. Detecting kCF

Contours are first extracted by the contour tracing method proposed in [2], which extracts 8-connected circular trajectories of black pixels that are adjacent to white pixels on the binary image. Key points which have a higher curvature on a contour are detected by the discrete contour evolution (DCE) approach [18] and the contour can be approximately represented by a polygon with these key points as vertices. We denote the detected key points as:

$$\vec{\mathbf{p}} = \{p_1, p_2, \ldots, p_T\} \quad (1)$$

where T is the number of vertices and can be controlled by a threshold in the DCE method. Fig. 3 shows an example of detected key points (the red points within the circles) on the contour of a connected component.

The method proposed in [30] collects contour fragments between every pair of key points on the shape contour. However, we think that the context around key points (which are high curvature points) contains useful information about the handwriting style. In order to maintain the informative context around key

Table 1
The number of documents in each key year of four cities in the MPS data set.

City	Key year											Sum
	1300	1325	1350	1375	1400	1425	1450	1475	1500	1525	1550	
Arnhem	72	115	22	30	52	73	78	38	36	27	42	585
Leiden	2	5	37	101	111	158	275	170	122	69	51	1101
Leuven	21	20	17	23	13	14	18	28	15	14	7	190
Groningen	2	3	15	20	56	81	138	187	200	132	148	982
Sum	97	143	91	174	232	326	509	423	373	242	248	2858

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

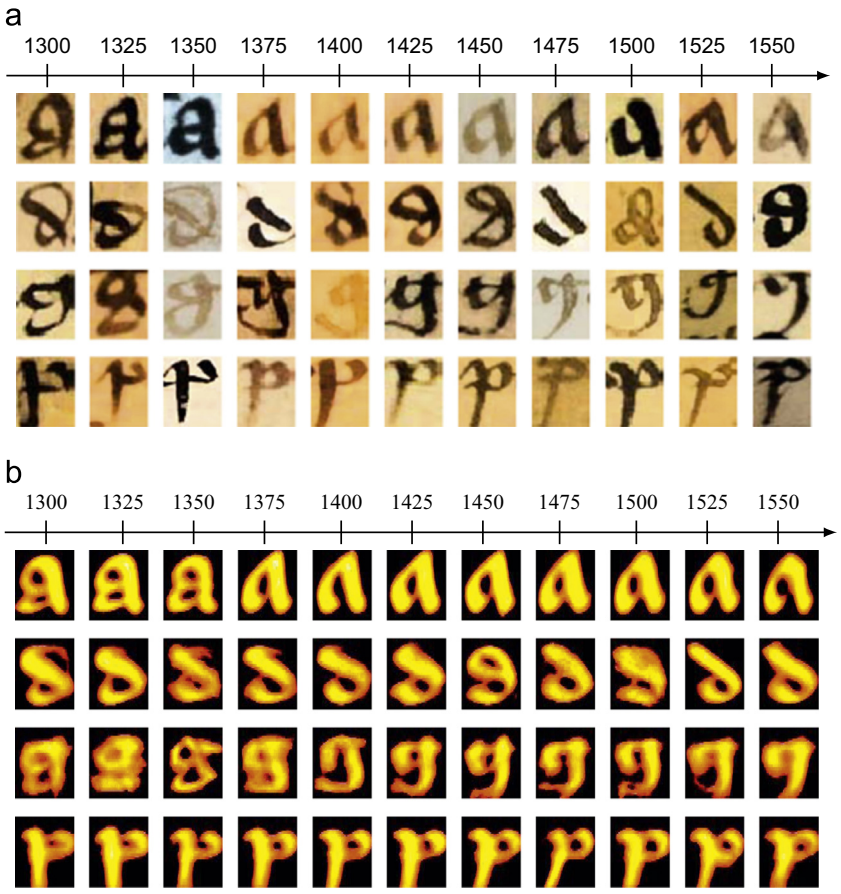

Fig. 2. (a) shows four labeled characters ("a", "d", "g", and "p" from top to bottom) in different key years in our MPS data set and (b) shows their models, defined as the average shapes of manually labeled characters in the Monk system [1]. While the individual allographes reveal style information, they miss the textural characteristic of samples, such as given in Fig. 1.

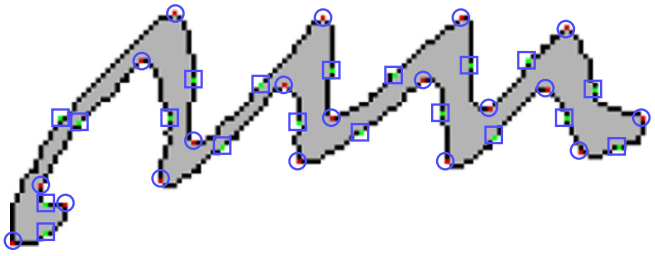

Fig. 3. A contour extracted on the connected component. The red points (with circle) are key points detected by the DCE [18] method and the green points (with rectangle) are the break points, necessary for capturing curvature information. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

points, we define break points $\vec{\mathbf{b}} = \{b_1, b_2, \ldots, b_T\}$ as the midpoints along the contour between two consecutive key points: the point b_i is the middle point on the contour fragment beginning at point p_i and end at point p_{i+1}. Fig. 3 shows an example of break points (the green points within the rectangles).

Given the contour and break points $\vec{\mathbf{b}} = \{b_1, b_2, \ldots, b_T\}$, primitive contour fragments can be obtained by segmenting the contour between pairs of consecutive break points (b_i, b_j), which are the short-range contour fragments. The long-range contour fragments can be obtained by concatenating k consecutive primitive contour fragments, which refers to k contour fragments (kCF). Fig. 4 shows kCF extracted from the contour in Fig. 3. From the figure we can see that as k grows, more and more complex and informative contour fragments can be obtained.

4.2. Describing kCF

It is important to develop a proper way to describe the detected informative kCF to facilitate comparing. The shape context [31] is used in [30] to describe contour fragments based on 5 reference points sampled equidistantly on the normalized contour fragments. However, determining the size of the shape context is arbitrary. In order to achieve the scale-invariant property, we use the relative coordinates of the fragment points as the feature vector, following the methods in [14,29]. Each contour fragment in a kCF is resampled such that it contains N_c coordinate points and then they are normalized to an origin of $(0, 0)$ and a standard deviation of radius 1 by:

$$\vec{x} \leftarrow (\vec{x} - \mu_x)/\sigma_x$$
$$\vec{y} \leftarrow (\vec{y} - \mu_y)/\sigma_y \quad (2)$$

where $\vec{x}$ and $\vec{y}$ are the collections of x and y coordinates of a contour fragment, μ_x and μ_y are averages of the $\vec{x}$ and $\vec{y}$ coordinates of the contour fragments and the σ_x and σ_y are the corresponding standard deviations. The final feature vector contains the normalized N_c $\vec{x}$ and $\vec{y}$ values and the dimension of the feature vector is $2N_c$.

There are two endpoints in each contour fragment (p_1 and p_2 in Fig. 5) and two feature vectors can be produced by starting at different endpoints. In order to make the final feature vector insensitive to the starting point, we carefully select the starting endpoint as follows. First, we find the midpoint $M = (x_m, y_m)$ of the contour fragment and the normalized distance of the pixels in

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

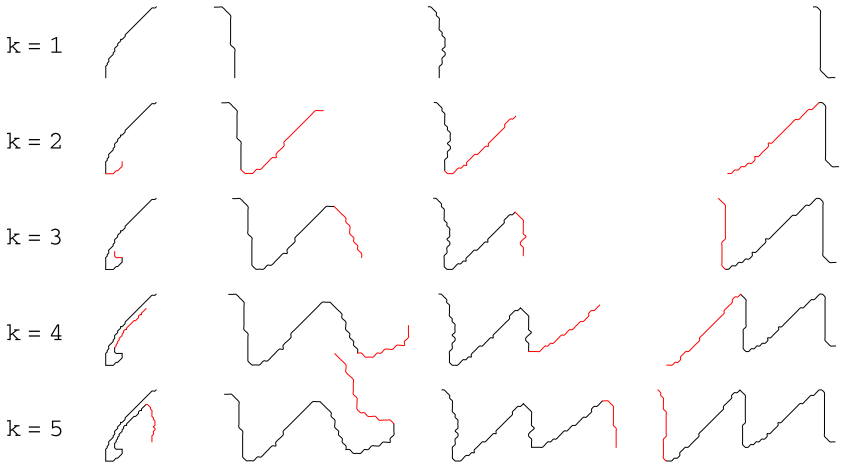

Fig. 4. Examples of contour fragments with different contour complexity degrees k extracted from the contour in Fig. 3. The red parts are the new added contour fragments when k grows. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

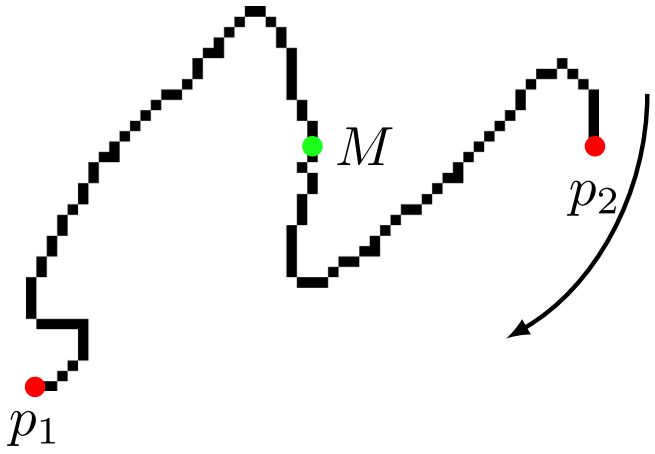

Fig. 5. An example of end point selection in a kCF. The red points p_1 and p_2 are two end points and the blue point m is the midpoint. We select the starting endpoint p_2 if $e_{p_2} < e_{p_1}$. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

each branch to the midpoint is given by:

$$e_{p_1} = \sum_{i=1}^{m} (|x_i| + |y_i|)$$
$$e_{p_2} = \sum_{i=m+1}^{N} (|x_i| + |y_i|) \quad (3)$$

where N is the number of points on the contour fragment. We select the starting endpoint p of the branch with the minimal value e_p.

Given a document from the MPS data set, we extract the contour fragments and use the proposed description method to represent the contour fragments. Fig. 6 shows four randomly selected contour fragments with 4CF and contour fragments on each row are found by the K nearest neighbor method with the Euclidean distance function, from which we can conclude that similar contour fragments may be from the same character or may be shared between different characters. Therefore the detected contour fragments can capture local contour structures and are informative and repeatable as well.

Our proposed method is different from the method proposed in [29], in which contour fragments with a specific length or number of points are extracted from contours, making the extracted contour fragments sensitive to image scaling. The proposed kCF is scale-invariant because key points detected by DCE are insensitive to scale changes. A connected component in historical documents may span several words or even several lines due to the touching strokes. Therefore, the CO^3 [14] extracted on these large connected components are sensitive to the touching strokes, making them non-repeatable. Our proposed kCF can solve such problem and is robust and more flexible than the CO^3.

4.3. Encoding kCF

The detected kCF can be considered as basic handwriting contours and the probability distribution of kCF can characterize the handwriting style. We construct codebooks for kCF with different k using clustering methods. It has been shown in [38] that the same performance was obtained for k-means,1D Kohonen self-organizing map (SOM) [39] and 2D SOM clustering methods. In this paper, we use the standard 2D SOM clustering method to train codebooks for kCF with Euclidean distance. Finally, one feature vector can be obtained for one document image and the dimension of the feature vector is determined by the size of the codebook.

5. *k* stroke fragments (*k*SF)

In general, handwritten characters are written by one or several strokes and the writing style can be represented by structures or shapes of strokes. In this section, we present three crucial steps to extract, describe and encode handwritten stroke fragments in document images.

5.1. Detecting kSF

In the literature, the term “stroke” in handwritten documents is used in slightly different ways. In on-line handwriting, strokes are determined by the velocity of the movement of the pen, or the writing speed [40]. In this case, strokes are “the pieces of handwriting movement bounded by minima in the tangential pen-tip velocity [41]”. That also means “a stroke is a trace of pen-tip movement which starts at pen-down and ends at pen-up [42]”. In order to provide clarity about the way the term “stroke” is used in this paper, we define the stroke in off-line handwritten documents as:

Definition 1. A stroke is a connected component of an ink trace which has two end points (one corresponds to the pen-down point and another to the pen-up point) on the stroke skeleton line.

One exception of this definition is the circle stroke, in which there are no end points (the skeleton line is also a circle). In order

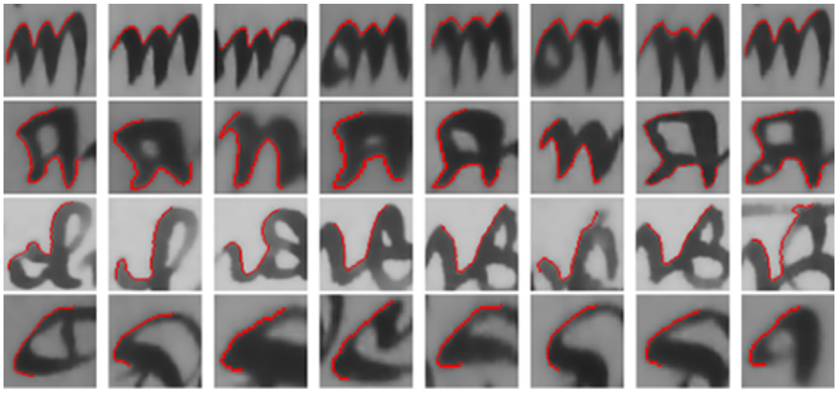

Fig. 6. A number of similar contour fragments with $k=4$ (4CF) detected in documents in the MPS data set. The red contours are the detected contour fragments. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.).

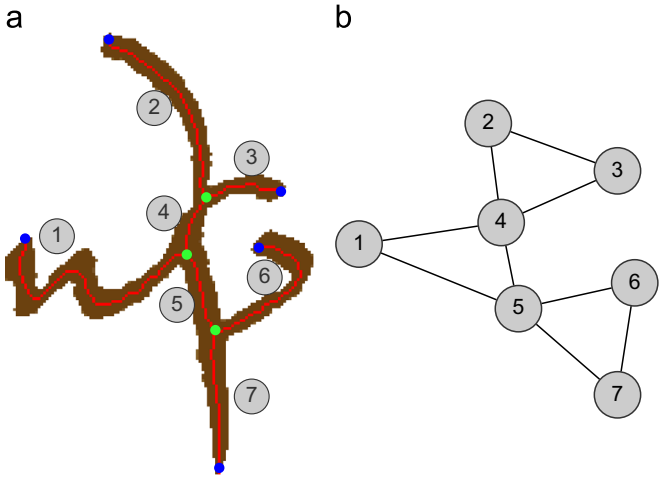

Fig. 7. (a) shows an example of a connected component in a historical document. The red line is the skeleton line of the ink, green points are the fork points and blue points are the end points. The connected component can be decomposed into seven parts segmenting at the fork points. (b) shows the corresponding stroke fragment graph (SFG). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

to integrate such circle strokes into our definition, we regard the left-most point in the skeleton line as the shared end points [14].

In a cursive handwritten document touching characters often form a large connected and complex structure and there is no obvious way to dissect it into stroke fragments. Fig. 7(a) gives an example of one connected component of the ink trace. The skeleton line of the connected component can be computed by thinning methods and there are two types of feature points on the skeleton line: *end points* and *fork points*. An *end point* refers to the beginning or end of a stroke (blue points in Fig. 7(a)), and a *fork point* (green points in Fig. 7(a)) is the location where at least two strokes meet [43]. Similar graph structures have been used for the temporal reconstruction of strokes from a static image [42].

In our paper, we consider fork points as the shared end points between touching strokes. Thus, the connected component can be decomposed into "strokes" segmenting at fork points, yielding stroke fragments between end points and fork points according to Definition 1 and these are called *primary stroke fragments*. For example, Fig. 7(a) shows a connected component with five end points and three fork points, and seven primary stroke fragments can be obtained, which are denoted by numbers 1–7. We refer to these stroke fragments as primitive stroke fragments because they are the minimal fragments which can be segmented from the connected component according to Definition 1.

This segmentation method is simple, intuitive and independent from any line detection or segmentation methods. However, it also yields fragments which are so small (especially the fragments between two fork points) that they become meaningless and can in some cases be regarded as noise (for example the 4th and 5th stroke fragments in Fig. 7(a)). In order to detect longer and more complex stroke fragments which are more informative, we build a stroke fragment graph (SFG) inspired by [44,45] as follows. Each node in the SFG corresponds to a primary stroke fragment and two nodes are linked if the two primary stroke fragments connect to each other, which means they share at least one fork point. Fig. 7(b) shows the SFG built from the primary stroke fragments in Fig. 7(a). The SFG reflects the relationship of connections between primitive stroke fragments of one connected component.

One important observation is that *any connected sub-graph in the SFG without loops corresponds to a stroke according to our stroke* Definition 1. For example, the sub-graph containing nodes {1, 4, 2} in the SFG in Fig. 7(b) can form a stroke which has two end points. In contrast, the sub-graph containing nodes {2, 3, 4}, which contains a loop, does not correspond to an effective stroke, because it has three end points and cannot be drawn in one time. We refer to strokes which contain a number of k primary stroke fragments (the length of the path between two vertexes in the SFG) as k stroke fragments or kSF. When $k=1$, 1SF are primitive stroke fragments. As k grows, more and more complex and informative strokes can be obtained. Fig. 8 gives an example of stroke fragments detected in the SFG in Fig. 7(b) when $k=3$ (3SF). In practice, given the value of k, all the connected paths without loops can be efficiently computed using the depth-first search method on the SFG.

5.2. Describing kSF

We use the polar stroke descriptor (PSD) proposed in our previous work [17] to describe kSF. The computation of the PSD is as follows: given a reference point $p_i=(x,y)$ and a direction ϕ, the distance from p_i to the ink boundary, called partial length $d_p(\phi)$, can be easily computed by searching the ink pixels following a ray in the direction ϕ [46]. A simple and efficient algorithm based on Bresenham's algorithm [47] is used to compute the distance from p_i to the ink boundary inspired by [2]. The end point $p_e=(x_e,y_e)$ is computed by

$$x_e = x + m * \cos(\phi)$$
$$y_e = y + m * \sin(\phi) \quad (4)$$

where the parameter m determines the maximum partial length or the maximum search space from p_i to p_e. An approximated linear path from p_i to p_e is constructed and the background point

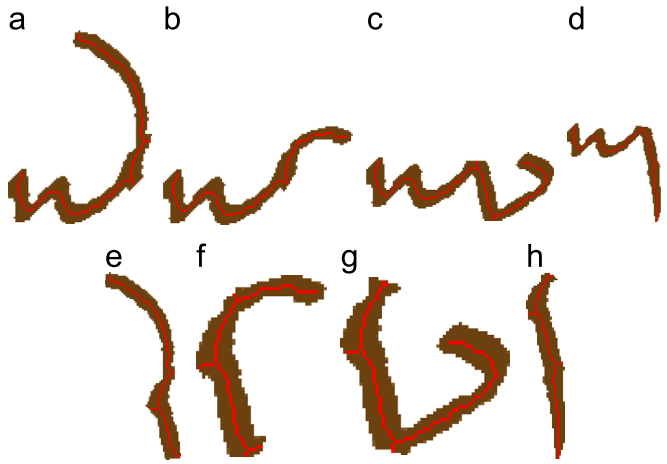

Fig. 8. Stroke fragments of 3SF generated in the SFG in Fig. 7(b). The corresponding nodes are: (a) {1,2,4}, (b) {1,3,4}, (c) {1,5,6}, (d) {1,5,6}, (e) {2,4,5}, (f) {3,4,5}, (g) {4,5,6}, and (h) {4,5,7}.

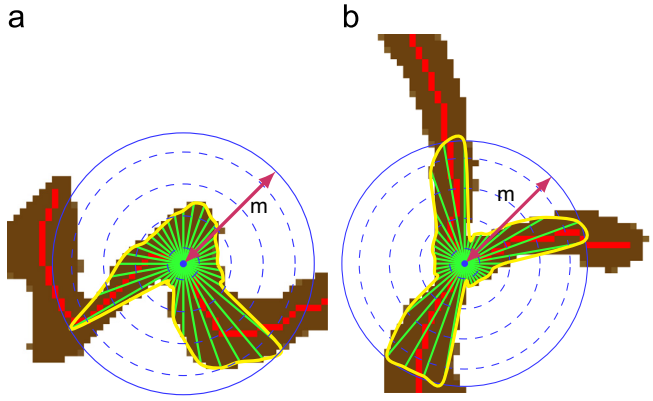

Fig. 9. An illustration of the polar stroke distribution on a reference point (the blue point in the center). The green rays are the partial length in each direction, and the yellow curve is the distribution of the partial length in the polar space. The red line is the skeleton line of the stroke ink. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

$p_b = (x_b, y_b)$ is found by tracing points starting from p_i towards to the end point p_e. The partial length is measured using a simple Euclidean distance:

$$d_p(\phi) = \sqrt{(x-x_b)^2 + (y-y_b)^2} \tag{5}$$

(More details of the computation of $d_p(\phi)$ can be found in [2,16].)

A partial length distribution is built on the reference point p_i by computing the partial length in every direction ϕ in a discrete set $\mathcal{D} = \{2\pi k/N; k = 0, \ldots, N-1\}$, where N is the number of directions we consider. This distribution is considered as the PSD of the point p_i, which is a local descriptor. Fig. 9 shows two examples of the PSD descriptors on the reference points in stroke fragments. Finally, the descriptor is normalized in order to make it scale-invariant.

The PSD is a rich descriptor, especially when the reference points lie on the fork points. In this case, it reflects the junction structure information in handwritten strokes, such as the radius and the number of branches of the junction region [48] (see example of Fig. 9(b)) which can be used for junction detection in handwritten documents [16].

The features of each *k*SF are computed as follows: N_s reference points on the skeleton line of *k*SF are sampled equidistantly and described by the PSD descriptor. Finally, these N_s PSD descriptors are concatenated into one feature vector to describe the corresponding *k*SF. In principle, the large number of N_s leads to a rich descriptor. However, when the N_s is too larger, the descriptor contains too much redundant information and the dimension of the descriptor is also high which needs a lot of computational time. In practice, we suggest the $N_s \in [5, 10]$. Fig. 10 gives an example of this method with 5 sample points.

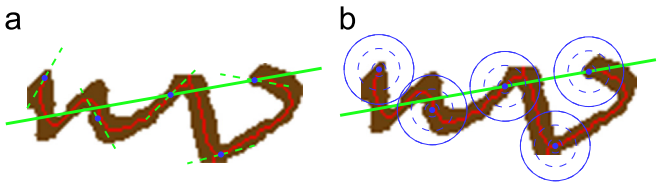

Fig. 10. (a) shows the sampled reference points (blue points) with tangent direction (dashed green line). The solid green direction is the estimated relative horizontal direction. (b) shows the PSD features (blue circles) on sampled points. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

In order to make *k*SF invariant to rotation, a relative horizontal direction should be used instead of the absolute horizontal direction in order to construct the PSD feature on each sampled point. The relative horizontal direction can be estimated by averaging the tangent angles of sampled points. Fig. 10 shows an example of the estimated relative direction.

Fig. 11 shows a number of stroke fragments with $k=1$ (1SF), which is also known as *Strokelets* [17]. Similar to *k*CF, *k*SF are also informative and repeatable and can be considered as mid-level representations.

As a grapheme-based method, our proposed *k*SF has several advantages: (1) Compared to the Junclets [16], the *k*SF captures the stroke properties in a large area and can be considered as a macro mid-level feature. (2) Compared to the Fraglets [15], our proposed *k*SF is easy to compute. Most importantly, the *k*SF is a script-independent grapheme-based method which can be used in any script. The descriptor of the *k*SF reflects the stroke properties, such as stroke width and stroke structures, which are lost in other methods [14–16,26].

5.3. Encoding kFS

In order to build a global feature representation for a historical document image, all *k*SF extracted from the image are mapped into a common space (named codebook) using the bag-of-words model [19]. As discussed in [15], there is no difference that existed between the performance of the codebooks trained by *k*-means, Kohonen SOM 1D and Kohonen SOM 2D. Similar to *k*CF, we use the Kohonen SOM 2D method [39] to train the codebook.

6. Experiments

6.1. Experimental settings

In the computation of the *k*CF and *k*FS, a binary method is needed to obtain the binary document image and compute contours and skeleton lines of the ink traces. Although several binarization methods have been proposed in the literature, such as [49–53], we apply the simple and efficient Otsu threshold algorithm [54] in our experiments, followed by the guided filter [55] to remove noise and make contours smooth. Each contour fragment of *k*CF is resampled to contain 100 points and the feature dimension is $100 \times 2 = 200$. The number of directions of the polar stroke descriptor (PSD) N is set to 120, which is the dimension of the PSD. In this paper, 10 points are sampled on each stroke fragment and each point is described by a PSD. Therefore, the dimension of *k*FS is $120 \times 10 = 1200$.

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

Fig. 11. A number of similar stroke fragments with $k=1$ (1SF) detected in documents in the MPS data set. The red lines are the skeleton lines and white points are the sampled reference points of PSD descriptors. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.).

We employed two widely used measures for performance evaluation: the mean absolute error (MAE) and cumulative score (CS) [56]. The MAE is a Manhattan-type distance, which is typically defined as:

$$MAE = \sum_{i=1}^{N} |\overline{K(y_i)} - K(y_i)| / N \tag{6}$$

where $K(y_i)$ is the ground-truth of the input document y_i and $\overline{K(y_i)}$ is the estimated key year, while N is the number of test documents. The cumulative score (CS) is typically defined as [56]:

$$CS(\alpha) = N_{e \le \alpha}/N \times 100\% \tag{7}$$

where $N_{e \le \alpha}$ is the number of test images on which the key year estimation makes an absolute error e no higher than the acceptable error level: α years. For historians, an error of ± 25 is, more often than not, acceptable when dating historical documents. Therefore, we report the cumulative score with error level $\alpha=25$ years in the experiments.

6.2. Historical document dating by general handwriting style identification

As we mentioned before, writing charters in the Middle Ages was a profession and the number of scribes simultaneously active in each city was limited. Therefore, an undated document can be dated by identifying the writer. This is reasonable because if we know the writer and his active period, the date of the document can be directly obtained [3,4]. We conduct experiments on writer identification on the MPS data set as well as historical document dating by handwriting style identification.

The writers of some charters are known in MPS and others are not. We term the subset of documents with writers who produced as least two samples as MPS-writer known with multiple samples (MPS-WKM for short) in which 143 writers produced 1127 documents, and term the subset of documents with writers who produced only one sample as MPS-writer known with single sample (MPS-WKS for short) and the rest of the documents without writer labels as MPS-writer unknown (MPS-WU for short) which contains 899 document images.

We perform writer identification on the MPS-WKM data set with χ^2 difference using the K nearest neighbors (KNN) method, following [15,26,16]. We utilize the "leave-one-out" strategy which is widely used for writer identification: taking the query document out and sorting the rest of the documents according to the distance function to output a hit list. The query document is recognized as the writer of the document on the top x of the hit list, corresponding to the top-x performance. Usually, the Top-1 and Top-10 performances are reported.

We also carry out historical document dating by general handwriting style identification. The combined MPS-WKM and MPS-WKS data sets with writer labels are considered as the reference data set. For each undated document in the MPS-WU data set, we find the K nearest neighbors using KNN in the reference data set and we assign the year to the undated document as the most represented years within the K nearest neighbors.

6.2.1. Performance of writer identification and dating

In this section, we present the performance of our proposed methods for writer identification and dating. We explore the degrees of complexity $k \in \{2,3,4,5\}$ for kCF and $k \in \{1,2,3\}$ for kSF. We do not consider 1CF because they contain less discriminative information as their lengths are too small. The feature dimensions of kCF and kSF are discussed in Section 6.3.2. Table 2 shows the performance of kCF and kSF for writer identification and dating, as well as Hinge [15], Quill [2] and Junclets [16], from which we can conclude that the writer identification rates increase for kCF while they decrease for kSF when k grows. A similar trend can be found for the dating performance. The writer identification performances of kSF are better than kCF, except 3SF and 5C, while the dating performances of kSF are worse than kCF, for all k. We can also find that Hinge achieves the best performance for writer identification and 3CF achieves the best performance for dating.

One interesting observation is that writer identification performances of kCF are worse than with all other features (except 3SF), while its dating performances are better than all other ones. The Hinge feature achieves the best performance for writer identification, while the dating performance is worse than Junclets, kCF (k=2,3,4,5) and kSF(k=1,2). We can obtain the conclusion that: *Features which achieve a good performance on writer identification are not necessarily suitable for historical document dating via writer identification when there exists no sample for a target writer in the training set.* The main reason is that dating requires features to capture the *general writing style in a certain period whereas writer identification needs features to capture the writing style characteristic for individuals precisely*.

From Table 2 we can also find that for features which are good in writer identification, the dating performance increases when K of KNN decreases, such as in the Hinge, Quill, Junclets, 1SF and 2SF features. However, for kCF, the best dating performances are mostly achieved when $K=20$.

In practice, we have found that combining the kCF and kSF do not improve the performance for both writer identification and dating. Therefore, their performances are not reported in this paper.

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

Table 2
The performance of writer identification and dating by handwriting style identification in terms of MAEs and CS(α=25) of the *k*CF, *k*SF and other features.

Method	Writer identification		Dating by writer identification (KNN)							
			K=5		K=10		K=20		K=50	
	Top-1	Top-10	MAEs	CS(α=25) (%)	MAEs	CS(α=25) (%)	MAEs	CS(α=25) (%)	MAEs	CS(α=25) (%)
Quill [2]	61.7	82.2	45.1	60.0	45.9	59.6	48.6	54.9	52.3	50.5
Hinge [15]	**71.8**	**85.9**	30.3	68.5	30.6	66.9	32.9	64.2	34.4	62.0
Junclets [16]	59.9	79.3	27.4	73.6	25.6	73.6	27.9	70.2	32.7	64.0
2CF	37.6	73.6	22.9	76.3	22.2	77.3	21.1	78.3	22.1	78.5
3CF	42.9	77.9	**18.7**	**80.9**	**18.4**	**80.9**	**17.9**	81.0	19.5	79.4
4CF	45.3	77.9	20.4	78.4	18.8	80.9	19.5	79.6	**19.4**	**79.5**
5CF	48.6	78.2	19.8	80.0	18.5	80.9	18.0	**81.6**	19.7	78.9
1SF	64.3	84.6	26.0	73.2	26.3	71.6	30.3	68.2	34.6	63.5
2SF	56.6	78.8	27.5	73.3	27.4	71.7	29.0	69.8	33.6	63.8
3SF	47.6	71.3	36.8	63.7	35.6	63.6	38.6	59.0	39.8	57.1

6.3. Historical document dating by classification

The dating problem can be considered as either a classification or a regression problem. In this paper, we regard it as a classification problem because the document distribution in our data set over the period of 1300–1550 CE has an obvious border between nearby key years. All the documents from each key year form a class and there are 11 classes which correspond to the 11 key years in the MPS data set. We train 11 corresponding classifiers using a linear SVM with a one-versus-all strategy and the undated document is assigned to the key year which has the maximum value of the 11 softmax output scores. The parameter C of the linear SVM is estimated by a grid search method. We split the data set into training (70%) and testing (30%) sets. The experiment is repeated 20 times and the average results are reported together with the standard deviation in the following experiments.

We consider two different evaluation scenarios for historical document dating. In the first one, we carefully split the data set into training and testing subsets to make sure that the same writer never appears in both training and test sets, which means that all documents from the same hand should be only in the training set or only in the test set. For documents without writer labels, we randomly split them into the training and test set. We term this scenario as *excluding writer duplicates* or *wr.excl.* for short. In the second scenario, we randomly split the data set into training and test sets without considering writer labels. We term this scenario as *including writer duplicates* or *wr.incl.* for short. In the *wr.excl.* scenario, the system performs the dating based on the general writing style built by other writers. However, in the *wr.incl.* scenario, the processing of writer identification is probably involved in the dating.

6.3.1. Performance of kCF and kSF

Table 3 shows the performance of historical document dating in terms of MAEs and CS(α=25) of the *k*CF and *k*SF in the *wr.excl.* and *wr.incl.* scenarios. The codebook sizes of *k*CF and *k*SF are set to 50×50 and 30×30, respectively. The selection of sizes are discussed in the next section. From the table we can find that for *k*CF, the MAEs decreases when k increases and the 5CF performs best. The MAE of 5CF is lower than 2CF by 5 and 4.4 years in the *wr.excl.* and *wr.incl.* scenarios, respectively. The same trend is also found in terms of CS(α=25) and 82.8 $\pm$ 3.6% documents are correctly estimated with error level no higher than 25 years in the *wr.excl.* scenario and the corresponding percentage in the *wr.excl.* scenario is 88.4 $\pm$ 1.6%. The results demonstrate that *k*CF with a higher k in a certain range offer informative, repeatable and discriminative contour fragments which capture the handwriting style in historical documents.

Table 3
MAEs and CS(α=25) of the *k*CF and *k*SF.

Method	*wr.excl.* scenario		*wr.incl.* scenario	
	MAEs	CS(α=25) (%)	MAEs	CS(α=25) (%)
2CF	26.7 $\pm$ 3.9	76.0 $\pm$ 4.3	17.3 $\pm$ 1.2	84.2 $\pm$ 1.9
3CF	23.8 $\pm$ 2.1	80.9 $\pm$ 2.4	14.3 $\pm$ 1.0	87.8 $\pm$ 1.5
4CF	22.8 $\pm$ 2.7	80.7 $\pm$ 3.8	13.3 $\pm$ 1.1	87.9 $\pm$ 1.5
5CF	**21.7** $\pm$ 2.8	**82.0** $\pm$ 3.6	**12.9** $\pm$ 1.1	**88.4** $\pm$ 1.6
1SF	22.1 $\pm$ 2.9	79.8 $\pm$ 3.1	12.6 $\pm$ 0.8	88.3 $\pm$ 1.1
2SF	**18.9** $\pm$ 2.0	**84.3** $\pm$ 3.0	**11.1** $\pm$ 0.8	**90.1** $\pm$ 1.4
3SF	23.8 $\pm$ 3.0	78.9 $\pm$ 3.0	15.1 $\pm$ 0.8	85.7 $\pm$ 1.3

From the results of the three degrees of *k*SF complexity in Table 3 we find that 2SF performs best overall. The average MAEs of the 2SF are 18.9/11.1 (for the *wr.excl./wr.incl.* scenarios) versus 22.1/12.6 and 23.8/15.1 of 1SF and 3SF, respectively. The CS(α=25) scores of 2SF in the two scenarios are also higher than the ones of 1SF and 3SF. The following order can be obtained: 2SF > 1SF > 3SF, by ranking *k*SF according to the average MAEs and CS(α=25) scores. The performance of 3SF is even worse than 1SF and the reason may be that 3SF contains too much artificial stroke fragments (see Fig. 8).

From Table 3 we also find that the performance of 2SF is better than 5CF by 2.8 and 1.8 years in terms of MAEs in the *wr.excl.* and *wr.incl.* scenarios, respectively. The descriptors of *k*SF do not only contain the curvature information of strokes, but also the stroke length distribution which reflects the stroke width and stroke distribution around sample points and the informative and discriminative information contained in the stroke fragments can be found by SVM.

6.3.2. The effect of codebook size

In this section, we conduct experiments to evaluate the performance of historical document dating by classification with different sizes of codebooks of the *k*CF and *k*SF. Figs. 12 and 13 show the results of the *k*CF and *k*SF, respectively. The two figures show that the MAEs of both *k*CF and *k*SF decrease as the size of the codebook increases.

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

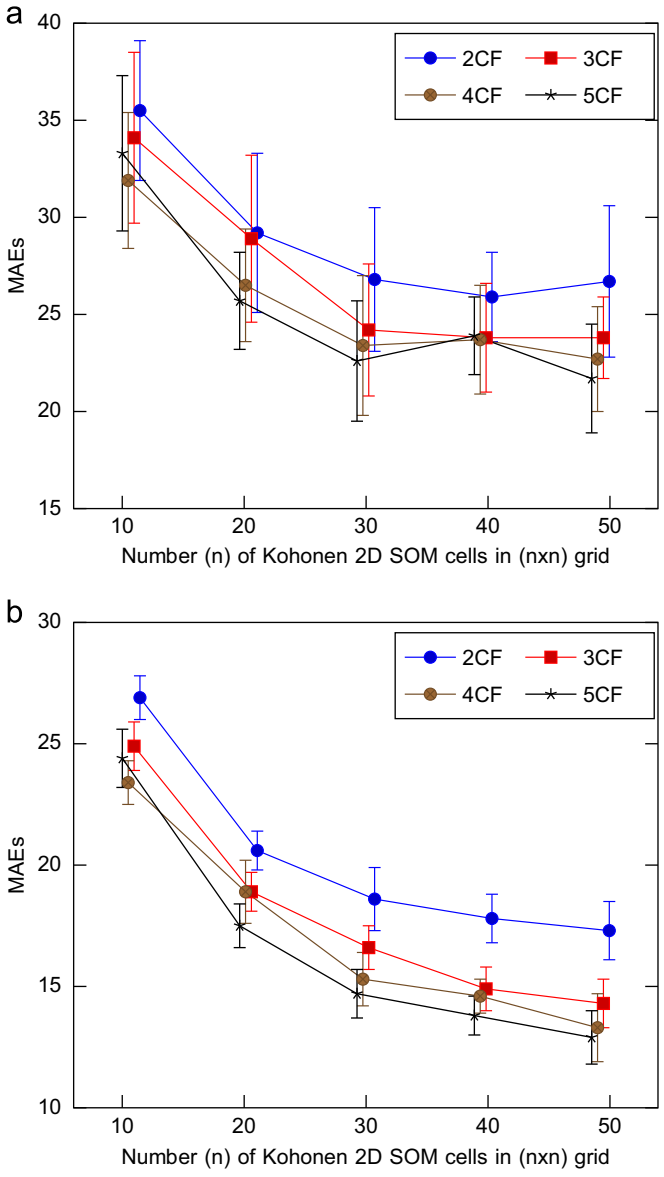

Fig. 12. The MAEs of *k*CF ($k=2,3,4,5$) with different codebook sizes. Note that the ranges of the *MAEs* axes are different between two figures in order to make them clear.

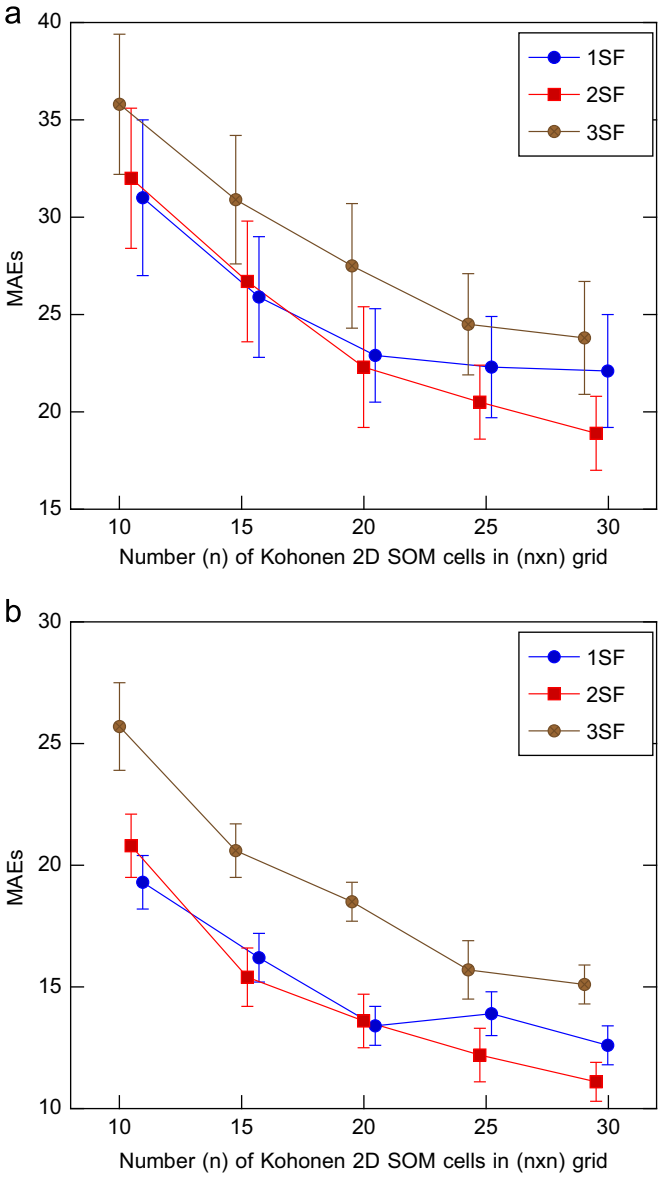

Fig. 13. The MAEs of *k*SF ($k=1,2,3$) with different codebook sizes. Note that the ranges of the *MAEs* axes are different between two figures in order to make them more clear.

Fig. 12(a) shows the performance of *k*CF with $k=2,3,4,5$ in the *wr.excl.* scenario. The best performances are achieved for *k*CF with a codebook size of 50×50, except the 2CF with 40×40. Fig. 12 (b) shows the MAEs of *k*CF with $k=2,3,4,5$ in the *wr.incl.* scenario and the lowest MAEs are obtained when the codebook size is 50×50. Therefore, the size of the codebook of *k*CF is set to 50×50 for $k=2,3,4,5$ in both the *wr.excl.* and the *wr.incl.* scenarios in the following experiments.

Similarly, Fig. 13(a) and (b) shows the MAEs of *k*SF ($k=1,2,3$) in the *wr.excl.* and *wr.incl.* scenarios, respectively. From the two figures we can find that the best performances are achieved with a codebook size of 30×30.

6.3.3. Performance of combined kCF and kSF

In this section, we evaluate performances when using several degrees of *k*CF and *k*SF simultaneously in the feature space. Table 4 gives the results of combined *k*CF and *k*SF in both the *wr.excl.* and *wr.incl.* scenarios. Generally, the *k*CF and *k*SF combined achieve better results than each *k* of the *k*CF and *k*SF separately. In the *wr. excl.* scenario, the {2345}CF achieves the lowest MAE (19.2 years),

Table 4
MAEs and CS($\alpha=25$) scores of *k*CF and *k*SF combined.

Method	*wr.excl.* scenario		*wr.incl.* scenario	
	MAEs	CS($\alpha=25$) (%)	MAEs	CS($\alpha=25$) (%)
(2+3)CF	22.9 ± 3.2	80.8 ± 3.2	14.2 ± 0.9	87.4 ± 1.8
(3+4)CF	22.4 ± 3.3	81.7 ± 3.5	12.1 ± 0.9	89.4 ± 1.5
(4+5)CF	20.3 ± 2.9	83.4 ± 3.3	11.8 ± 0.8	89.9 ± 1.5
(2+3+4)CF	21.5 ± 3.1	82.4 ± 4.5	12.0 ± 0.8	89.2 ± 1.3
(3+4+5)CF	20.0 ± 2.9	83.6 ± 3.2	$\mathbf{10.7}\pm1.1$	90.5 ± 1.9
(2+3+4+5)CF	$\mathbf{19.2}\pm3.5$	$\mathbf{85.8}\pm2.8$	10.8 ± 0.9	$\mathbf{90.8}\pm1.1$
(1+2)SF	18.6 ± 2.3	84.5 ± 3.6	10.1 ± 0.7	91.2 ± 1.3
(1+2+3)SF	$\mathbf{17.4}\pm1.9$	$\mathbf{86.8}\pm2.0$	$\mathbf{9.9}\pm0.6$	$\mathbf{91.8}\pm1.5$
(1+2+3)SF+ (2+3+4+5)CF	$\mathbf{14.9}\pm1.7$	$\mathbf{89.2}\pm2.4$	$\mathbf{7.9}\pm1.0$	$\mathbf{93.2}\pm1.3$

which is better than other combinations. Although the best performance in term of MAE is obtained by {345}CF in the *wr.incl.* scenario, there is no obvious difference between the performance

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

Table 5
MAEs and CSs of the combination of other features with the proposed *k*CF and *k*SF.

Method	*wr.excl.* scenario		*wr.incl.* scenario	
	MAEs	CS(α=25) (%)	MAEs	CS(α=25) (%)
Quill [2]	23.7 ± 2.9	80.6 ± 3.0	12.1 ± 0.9	89.5 ± 1.3
Hinge [15]	22.1 ± 2.9	80.6 ± 3.1	12.2 ± 0.9	89.6 ± 1.3
Junclets [16]	21.5 ± 3.3	81.9 ± 3.9	12.0 ± 0.7	89.2 ± 1.4
(2+3+4+5)CF	19.2 ± 3.5	85.8 ± 2.8	10.8 ± 0.9	90.8 ± 1.1
(1+2+3)SF	17.4 ± 1.9	86.8 ± 2.0	9.9 ± 0.6	91.8 ± 1.5
(1+2+3)SF+(2+3+4+5)CF	**14.9** ± 1.7	**89.2** ± 2.4	**7.9** ± 1.0	**93.2** ± 1.3

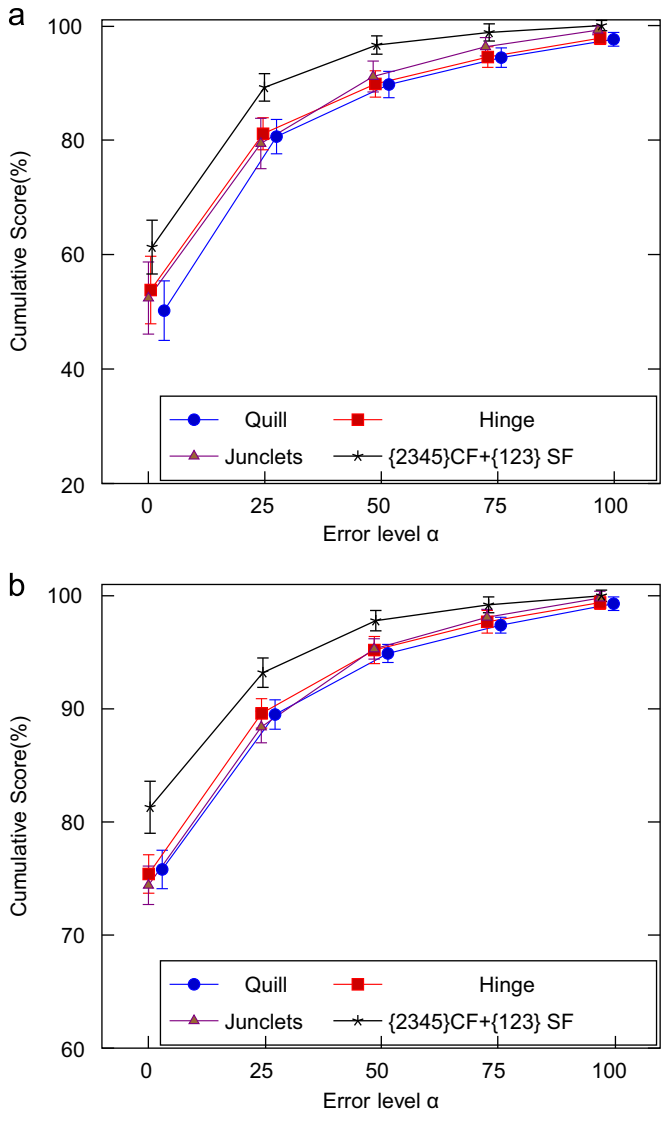

Fig. 14. CS curves of the error level from 0 to 100 years of different methods applied to the MPS data set. Note that the ranges of CS axes are different between two figures in order to make curves clear.

of {345}CF and {2345}CF and the CS(α=25) score of {2345}CF is higher than the one of {345}CF. Comparing the results of Table 4 with the ones of Table 3, we find that the combination of *k*CF improves the best performance of single *k*CF from 21.7 to 19.2 (MAE) and from 82.0% to 85.8% (CS(α=25)) in the *wr.excl.* scenario. Correspondingly, in the *wr.incl.* scenario, the best performance is improved from 12.9 to 10.7 (MAE) and from 88.4% to 90.8% (CS(α=25)).

Table 6
The dating accuracy (MAEs and CS(α=25)) on the MPS data set for different methods.

Method	MAE	CS(α=25)
Random guess	85.3 ± 58.5	25.7%
Monk [1]	36.0 ± 20.6	–
Study [20]	35.4	63.5%
Study [17]	20.9	–
(1+2+3)SF+(2+3+4+5)CF (*wr.excl.*)	14.9 ± 1.7	89.2 ± 2.4%
(1+2+3)SF+(2+3+4+5)CF (*wr.incl.*)	7.9 ± 1.0	93.2 ± 1.3%

Although the performance of 3SF is worse than 1SF and 2SF, combining it with {12}SF achieves the best results, which demonstrates that 3SF can provide some useful information discovered by SVM. Comparing Table 4 with Table 3, the MAEs and CS (α=25) in the *wr.excl.* and *wr.incl.* scenarios are improved by 1.5/2.5%, 1.2/1.7%, respectively.

We also combine {2345}CF and {123}SF together and the results are shown in the bottom row of Table 4. The combined performance outperforms all individual features ({2345}CF and {123}SF) involved in the combination. The MAEs of the combined {2345}CF and {123}SF are 14.9 and 7.9 in the *wr.excl.* and *wr.incl.* scenarios, respectively, which are the best ones among all the combinations. The results demonstrate that the *k*CF and *k*SF capture different types of information about handwriting styles and combining them can improve performance.

6.3.4. Comparison with other features

In Table 5, we present the performances of other existing features, such as the Quill [2], Hinge [15] and Junclets [16]. From Table 5 we can see that the performances of {2345}CF, {123}SF and the combined {2345}CF and {123}SF are better than performances of Quill, Hinge and Junclets.

In practice, we have found that there is no significant difference between the combination of {2345}CF and {123}SF and the combination of {2345}CF and {123}SF with Quill, Hinge and Junclets. The main reason is that *k*CF captures curvature information of contours with Quill and Hinge that is similar to the stroke structures captured by *k*SF with Junclets. In fact, *k*SF contains junction information because we consider fork points as the shared end points and descriptors of these end points are included in *k*SF. Furthermore, the proposed *k*CF and *k*SF are more flexible and insensitive to the scale and rotation transform. Fig. 14 shows the CS curves of Quill, Hinge and Junclets and the proposed {2345}CF and {123}SF combined. From the figure we can find that the CS curve of our proposed method is above that of Quill, Hinge and Junclets and our proposed method improves performance, especially when the error level is small ($\alpha \leq 50$).

6.3.5. Comparison with other studies

In this section, we summarize all our results with historical document dating and compare them with a random guess in Table 6. Manually labeled characters were used for dating in the Monk [1] system and the results can be found on our website.[1] In [20], a global regression method with a local adjust regression was used to estimate the key year of document images based on the combined Hinge and Fraglets features. Later in [17], the strokelets (1SF), which is also a special case of *k*SF, were used with the KNN method. Although it is not entirely fair to compare them because many document images have been added to the MPS data set for this paper and some low quality images were replaced by high

[1] http://application02.target.rug.nl/monk/Overslag/date-histogram-MPS.html

quality images, Table 6 still shows the improvement achieved by our proposed method.

7. Discussion and conclusion

We have introduced the *k*CF and *k*SF family of contour and stroke fragment features and applied them to historical document dating based on the MPS data set. The *k*CF and *k*SF are scale and rotation invariant grapheme-based features which can capture the handwriting style of handwritten documents. We approached dating in two ways: by handwriting style identification and by classification. Concerning dating by handwriting style identification, we found that features which achieve good performances for writer identification are not suitable for historical document dating by handwriting style identification by means of writer identification when there is no duplicated document existed in the training set. For example, *k*CF performed worse for writer identification than other methods but better than others for dating.

As far as dating by classification is concerned, we evaluated the performance of the proposed *k*CF and *k*SF in two scenarios: excluding writer duplicates (*wr.excl.*) and including writer duplicates (*wr.incl.*) and experimental results demonstrated that a combination of *k*CF and *k*SF achieves state-of-the-art results on the MPS data set. Several interesting conclusions can be drawn from our experimental results. First, the performance of *k*CF increases with an increasing complexity *k*. However, with a large *k*, the *k*CF may contain long contour fragments which are not informative or repeatable in the document images. This is also true for *k*SF and 2SF performs better than either 1SF or 3SF. Secondly, *k*CF and *k*SF contain different information. *k*CF captures the curvature information under different scales which contains both local (small *k*) and intermediate (large *k*) contour information of the handwriting style, while *k*SF captures the stroke structure caused by both the writing instrument and handwriting style. Therefore, only by combining them we achieved an optimal performance.

There are some specific requirements for paleographers and historians when solving the dating problem by means of the computer. For example, how to visualize the informative patterns which correlate with temporal information and how to discover the development or evolution of handwriting styles instead of determining dates by classification or regression. Our proposed *k*CF and *k*SF are grapheme-based features which can be visualized and our future tasks include designing an interface for end users.

Conflict of interest

None declared.

Acknowledgments

This work has been supported by the Dutch Organization for Scientific Research NWO (Project no. 380-50-006).

References

[1] T. Van der Zant, L. Schomaker, K. Haak, Handwritten-word spotting using biologically inspired features, IEEE Trans. Pattern Anal. Mach. Intell. 30 (11) (2008) 1945–1957.
[2] A. Brink, J. Smit, M. Bulacu, L. Schomaker, Writer identification using directional ink-trace width measurements, Pattern Recognit. 45 (1) (2012) 162–171.
[3] M. Panagopoulos, C. Papaodysseus, P. Rousopoulos, D. Dafi, S. Tracy, Automatic writer identification of ancient Greek inscriptions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (8) (2009) 1404–1414.
[4] D. Arabadjis, F. Giannopoulos, C. Papaodysseus, S. Zannos, P. Rousopoulos, M. Panagopoulos, C. Blackwell, New mathematical and algorithmic schemes for pattern classification with application to the identification of writers of important ancient documents, Pattern Recognit. 46 (8) (2013) 2278–2296.
[5] C. Papaodysseus, P. Rousopoulos, F. Giannopoulos, S. Zannos, D. Arabadjis, M. Panagopoulos, E. Kalfa, C. Blackwell, S. Tracy, Identifying the writer of ancient inscriptions and Byzantine codices. A novel approach, Comput. Vis. Image Underst. 121 (2014) 57–73.
[6] J.-P. Van Oosten, L. Schomaker, Separability versus prototypicality in handwritten wordimage retrieval, Pattern Recognit. 47 (3) (2014) 1031–1038.
[7] M. Rusiñol, D. Aldavert, R. Toledo, J. Lladós, Efficient segmentationfree keyword spotting in historical document collections, Pattern Recognit. 48 (2) (2015) 545–555.
[8] N.R. Howe, S. Feng, R. Manmatha, Finding words in alphabet soup: inference on freeform character recognition for historical scripts, Pattern Recognit. 42 (12) (2009) 3338–3347.
[9] J. Richarz, S. Vajda, R. Grzeszick, G.A. Fink, Semi-supervised learning for character recognition in historical archive documents, Pattern Recognit. 47 (3) (2014) 1011–1020.
[10] F. Palermo, J. Hays, A.A. Efros, Dating historical color images, in: Computer Vision—ECCV, 2012, pp. 499–512.
[11] Y.J. Lee, A.A. Efros, M. Hebert, Style-aware mid-level representation for discovering visual connections in space and time, in: International Conference on Computer Vision (ICCV), 2013, pp. 1857–1864.
[12] X. Geng, Z.-H. Zhou, K. Smith-Miles, Automatic age estimation based on facial aging patterns, IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) (2007) 2234–2240.
[13] G. Guo, Y. Fu, C.R. Dyer, T.S. Huang, Image-based human age estimation by manifold learning and locally adjusted robust regression, IEEE Trans. Image Process. 17 (7) (2008) 1178–1188.
[14] L. Schomaker, M. Bulacu, Automatic writer identification using connected-component contours and edge-based features of uppercase western script, IEEE Trans. Pattern Anal. Mach. Intell. 26 (6) (2004) 787–798.
[15] M. Bulacu, L. Schomaker, Text-independent writer identification and verification using textural and allographic features, IEEE Trans. Pattern Anal. Mach. Intell. 29 (4) (2007) 701–717.
[16] S. He, M. Wiering, L. Schomaker, Junction detection in handwritten documents and its application to writer identification, Pattern Recognit. 48 (12) (2015) 4036–4048.
[17] S. He, L. Schomaker, A polar stroke descriptor for classification of historical documents, in: International Conference on Document Analysis and Recognition (ICDAR), 2015, pp. 6–10.
[18] L.J. Latecki, R. Lakämper, Convexity rule for shape decomposition based on discrete contour evolution, Comput. Vis. Image Underst. 73 (3) (1999) 441–454.
[19] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: Workshop on Statistical Learning In Computer Vision (ECCV), 2004.
[20] S. He, P. Samara, J. Burgers, L. Schomaker, Towards style-based dating of historical documents, in: International Conference of Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 265–270.
[21] M. Bulacu, L. Schomaker, L. Vuurpijl, Writer identification using edge-based directional features, in: International Conference on Document Analysis and Recognition (ICDAR), 2003, pp. 937–941.
[22] S. He, L. Schomaker, Delta-n hinge: rotation-invariant features for writer identification, in: International Conference on Pattern Recognition (ICPR), 2014, pp. 2023–2028.
[23] H.E. Said, T.N. Tan, K.D. Baker, Personal identification based on handwriting, Pattern Recognit. 33 (1) (2000) 149–160.
[24] B. Helli, M.E. Moghaddam, A text-independent Persian writer identification based on feature relation graph (FRG), Pattern Recognit. 43 (6) (2010) 2199–2209.
[25] A.J. Newell, L.D. Griffin, Writer identification using oriented basic image features and the delta encoding, Pattern Recognit. 47 (6) (2014) 2255–2265.
[26] I. Siddiqi, N. Vincent, Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features, Pattern Recognit. 43 (11) (2010) 3853–3865.
[27] M.N. Abdi, M. Khemakhem, A model-based approach to offline text-independent Arabic writer identification and verification, Pattern Recognit. 48 (5) (2014) 1890–1903.
[28] G. Zhu, X. Yu, Y. Li, D. Doermann, Language identification for handwritten document images using a shape codebook, Pattern Recognit. 42 (12) (2009) 3184–3191.
[29] G. Ghiasi, R. Safabakhsh, Offline text-independent writer identification using codebook and efficient code extraction methods, Image Vis. Comput. 31 (5) (2013) 379–391.
[30] X. Wang, B. Feng, X. Bai, W. Liu, L.J. Latecki, Bag of contour fragments for robust shape classification, Pattern Recognit. 47 (6) (2014) 2116–2125.
[31] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 24 (4) (2002) 509–522.
[32] C. Zhao, S.-F. Chan, W.-K. Cham, L.-M. Chu, Plant identification using leaf shapes—a pattern counting approach, Pattern Recognit. 48 (10) (2015) 3203-3215.
[33] F. Wahlberg L. Martensson A. Brun, Large scale style based dating of medieval manuscripts, in: Workshop on Historical Document Image and Processing (HIP), 2015, pp. 107–114.

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032

[34] N.R. Howe, A. Yang, M. Penn, A character style library for Syriac manuscripts, in: Workshop on Historical Document Image and Processing (HIP), 2015, pp. 123–128.
[35] Y. Li, D. Genzel, Y. Fujii, A. C.Popat, Publication date estimation for printed historical documents using convolutional neural networks, in: HIP, 2015, pp. 99–106.
[36] L. Vincent, Google book search: document understanding on a massive scale, in: International Conference on Document Analysis and Recognition (ICDAR), vol. 2, 2007, pp. 819–823.
[37] P. Samara, Towards a medieval palaeographical scale, in: I. Fees, et al. (Eds.), Papsturkundenforschung zwischen internationaler Vernetzung und Digitalisierung, Munich, 2014.
[38] M. Bulacu, L. Schomaker, A comparison of clustering methods for writer identification and verification, in: International Conference on Document Analysis and Recognition (ICDAR), 2005, pp. 1275–1279.
[39] T. Kohonen, Self-Organization and Associative Memory, Springer Verlag, 1989.
[40] L.R. Schomaker, H.-L. Teulings, A handwriting recognition system based on properties of the human motor system, in: International Workshop on Frontiers in Handwriting Recognition, Montreal, April, Citeseer, 1990.
[41] L. Schomaker, Using stroke-or character-based self-organizing maps in the recognition of on-line, connected cursive script, Pattern Recognit. 26 (3) (1993) 443–450.
[42] Y. Kato, M. Yasuhara, Recovery of drawing order from single-stroke handwriting images, IEEE Trans. Pattern Anal Mach. Intell. 22 (9) (2000) 938–949.
[43] K. Liu, Y.S. Huang, C.Y. Suen, Identification of fork points on the skeletons of handwritten Chinese characters, IEEE Trans. Pattern Anal. Mach. Intell. 21 (10) (1999) 1095–1100.
[44] V. Ferrari, T. Tuytelaars, L. Van Gool, Object detection by contour segment networks, in: Computer Vision—ECCV 2006, 2006, pp. 14–28.
[45] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, Groups of adjacent contour segments for object detection, IEEE Trans. Pattern Anal. Mach. Intell. 30 (1) (2008) 36–51.
[46] B. Epshtein, E. Ofek, Y. Wexler, Detecting text in natural scenes with stroke width transform, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2963–2970.
[47] D. Hearn, M.P. Baker, Computer Graphics, C version, vol. 2, Prentice Hall, Upper Saddle River, 1997.
[48] L. Parida, D. Geiger, R. Hummel, Junctions: detection, classification, and reconstruction, IEEE Trans. Pattern Anal. Mach. Intell. 20 (7) (1998) 687–698.
[49] B. Gatos, I. Pratikakis, S.J. Perantonis, Adaptive degraded document image binarization, Pattern Recognit. 39 (3) (2006) 317–327.
[50] M.A. Ramírez-Ortegón, E. Tapia, L.L. Ramírez-Ramírez, R. Rojas, E. Cuevas, Transition pixel: a concept for binarization based on edge detection and gray-intensity histograms, Pattern Recognit. 43 (4) (2010) 1233–1243.
[51] R.F. Moghaddam, M. Cheriet, A multi-scale framework for adaptive binarization of degraded document images, Pattern Recognit. 43 (6) (2010) 2186–2198.
[52] R.F. Moghaddam, M. Cheriet, AdOtsu: an adaptive and parameterless generalization of Otsu's method for document image binarization, Pattern Recognit. 45 (6) (2012) 2419–2431.
[53] M.A. Ramírez-Ortegón, L.L. Ramírez-Ramírez, V. Märgner, I.B. Messaoud, E. Cuevas, R. Rojas, An analysis of the transition proportion for binarization in handwritten historical documents, Pattern Recognit. 47 (8) (2014) 2635–2651.
[54] N. Otsu, A threshold selection method from gray-level histograms, Automatica 11 (285–296) (1975) 23–27.
[55] K. He, J. Sun, X. Tang, Guided image filtering, IEEE Trans. Pattern Anal. Mach. Intell. 35 (6) (2013) 1397–1409.
[56] X. Geng, Z.-H. Zhou, K. Smith-Miles, Automatic age estimation based on facial aging patterns, IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) (2007) 2234–2240.

Sheng He received his B.S. and M.S. degrees both from Northwestern Polytechnical University, Xi'an, China, in 2009 and 2012, respectively. He is currently pursuing the Ph.D. degree in the Artificial Intelligence Department, University of Groningen, The Netherlands. His research interests include pattern recognition, image processing and handwritten document analysis.

Petros Samara received his M.S. in philosophy from the Erasmus University, Rotterdam, in 2002 and his M.S. degree in medieval history from the Free University of Amsterdam in 2005. He is currently a visiting scholar at the Huygens Institute for the History of The Netherlands in The Hague, working on his dissertation on the development of late medieval documentary script in The Netherlands. His research interests include medieval palaeography and the history of philosophy.

Jan Burgers studied medieval history at the University of Amsterdam and received his Ph.D. degree in 1993, cum laude, on the palaeography of the documentary sources of Holland and Zeeland in the 13th century. At present he is a senior researcher at the Huygens Institute of the History of The Netherlands in The Hague and a part-time professor at the University of Amsterdam. He has produced over 120 publications on palaeography, diplomatics, medieval chronicles, as well as source editions.

Lambert Schomaker is a full professor in artificial intelligence at the University of Groningen and the director of its AI institute ALICE since 2001. His main interest is in pattern recognition and machine learning problems, with applications in handwriting recognition problems. He has contributed to over 150 peer-reviewed publications in journals and books (h=16/ISI, h=37/Google Citations). His work is cited in 23 patents. In recent years his focus is on continuous-learning systems and bootstrapping problems, where learning starts with very few examples. Prof. Schomaker is a senior member of IEEE, member of the IAPR and is a member of a number of Dutch research programme committees in e-Science (NWO), Computational Humanities (KNAW), computational science and energy (Shell/NWO/FOM). He received IBM Faculty Awards (2011, 2012) for the Monk word retrieval system in historical manuscript collections using high-performance computing.

Please cite this article as: S. He, et al., Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2016.03.032