A Multiple-Label Guided Clustering Algorithm for Historical Document Dating and Localization

Sheng He, Petros Samara, Jan Burgers, and Lambert Schomaker, *Senior Member, IEEE*

***Abstract*—It is of essential importance for historians to know the date and place of origin of the documents they study. It would be a huge advancement for historical scholars if it would be possible to automatically estimate the geographical and temporal provenance of a handwritten document by inferring them from the handwriting style of such a document. We propose a multiple-label guided clustering algorithm to discover the correlations between the concrete low-level visual elements in historical documents and abstract labels, such as date and location. First, a novel descriptor, called histogram of orientations of handwritten strokes, is proposed to extract and describe the visual elements, which is built on a scale-invariant polar-feature space. In addition, the multi-label self-organizing map (MLSOM) is proposed to discover the correlations between the low-level visual elements and their labels in a single framework. Our proposed MLSOM can be used to predict the labels directly. Moreover, the MLSOM can also be considered as a pre-structured clustering method to build a codebook, which contains more discriminative information on date and geography. The experimental results on the medieval paleographic scale data set demonstrate that our method achieves state-of-the-art results.**

***Index Terms*—Historical document dating, historical document localization, histogram of orientations of handwritten stroke, multi-label self-organizing map.**

I. Introduction

MANY visual elements of images of the visual world can be correlated to a symbolic label [1]. For example, visual elements from Google Street View [2] Images contain geographical information [3], visual elements of images of historical cars are correlated with temporal information [1] and visual elements in printed texts vary over different languages and scripts [4], [5]. Finding the corresponding visual elements common to different labels can reveal the subtle difference between categories. Therefore, discovering the correlations between the visual elements style and their labels is very useful for resolving computer vision problems, such as localization [1], [3], dating [1], [6], age estimation based on face images [7], image classification [8] and script and font identification [4], [5].

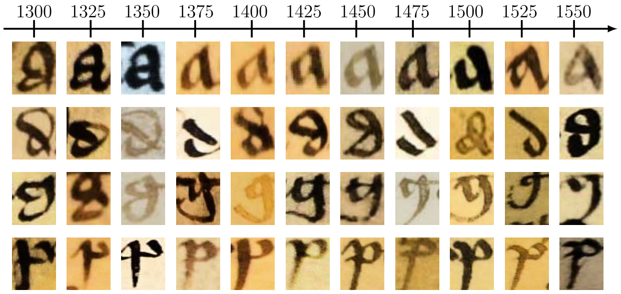


Fig. 1. Four labeled characters (‘a’,‘d’,‘g’,‘p’ from top to bottom) in different years in our MPS data set.

Knowing *when* and *where* historical sources were written is of course essential to scholars who study them. However, countless historical manuscripts are not dated and bear no indication of their place of origin. This is especially true for manuscripts originating from the Middle Ages. The process of dating and localizing a manuscript of unknown origin by inferring these information from the nature of the script it contains is usually not based on objective, measurable criteria, but on the individual non-verbal intuition of paleographical experts. This manual evaluation is often regarded as the prerogative of a few specialists, who, moreover, sometimes arrive at conflicting conclusions, without providing clearly verifiable arguments. The aim of this paper is to build a system which can automatically discover the correlations between the visual elements - which are predominantly patterns of strokes or sub-strokes - in document images and labels pertaining to year and origin. The discovered correlations can be used to date and localize unlabeled query documents.

Our basic assumption about historical document dating and localization is that writing styles undergo a gradual, continuous change. This is a general phenomenon which is also observable in other forms of dating based on historical images, such as photographs of different cars [1]. The writing style is reflected in the way characters were written, and the writing style of the same character written in different decades is therefore usually somewhat different. For example, four characters are shown in Fig. 1 written in different periods. Notice that the writing style of the character ‘a’ changes dramatically from 1375 to 1400 and keeps consistent from 1400 to 1550. Other characters show a subtle evolution from 1300 to 1550 (more

Manuscript received November 13, 2015; revised May 27, 2016 and July 21, 2016; accepted August 18, 2016. Date of publication August 24, 2016; date of current version September 16, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiao-Ping Zhang.

S. He and L. Schomaker are with the Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen, 9700 AK Groningen, The Netherlands (e-mail: heshengxgd@gmail.com; l.schomaker@ai.rug.nl).

P. Samara is with the Huygens Institute for the History of the Netherlands, 2509 LT The Hague, The Netherlands (e-mail: petros.samara@huygens.knaw.nl).

J. Burgers is with the Huygens Institute for the History of the Netherlands, 2509 LT The Hague, The Netherlands, also with the Royal Netherlands Academy of Arts and Sciences, 2509 LT The Hague, The Netherlands, and also with the Department of History, University of Amsterdam, 1012 CX Amsterdam,The Netherlands (e-mail:jan.burgers@huygens.knaw.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2602078

examples are shown in the Monk website [9]). Therefore, the writing styles of characters contain the temporal information studied by paleographic experts [10].

Several works have been proposed in the last decade in order to determine the "age" of images. For example, facial age estimation based on face images has been widely studied in [7], [11], and [12]. Automatically estimating the age of historical color photographs based on the extracted temporally discriminative information has been proposed in [13]. Another related work about discovering the visual element connections to time and space has been proposed in [1], specifically to solve the dating and localization problem. After finding the initial style-sensitive visual elements, this algorithm discovers the correspondences by incrementally revising the initial visual elements within "nearby" labels. The recognition of cities by means of a visual element mining method has been studied in [3], which automatically discovers the geographical image elements which are representative for a certain city. The discriminative patterns which are connected to certain parts or objects are discovered in [8] for image classification. For example, the patterns of the arches and the benches are usually connected to a *Church* and the patterns of the center table and the seats are usually connected a *Meeting Room*.

The historical document dating problem has been recently studied in [6] and [14]-[16]. Our previous work in [6] estimated the year of origin of historical documents using the MPS data set, utilizing the features that have been successfully used in writer identification. In [16], we used a family of local contour fragments(*k*CF) and stroke fragments (*k*SF) for dating. In [14], a stroke shape model has been used for medieval document dating, based on a collection of medieval charters from Sweden. The authors in [15] used inkball models for Syriac document dating on a collection of securely dated letter samples between 500 and 1100 CE.

In this paper, we propose a novel descriptor, Histogram of Orientation of Handwritten Stroke (HOHS or H_2OS), to extract and represent the visual elements (strokes or parts of the strokes) in historical documents. In contrast to existing features, such as the Histogram of Oriented Gradients (HOG) descriptor [17], our proposed H_2OS is a scale-invariant descriptor which uses the stroke width as the scale factor. A weakly-supervised Self-Organizing Map (SOM) [18] method is proposed to introduce the label information in the self-organization process to discover the relationships of visual elements in each label space. Our proposed method is called Multi-Label Self-Organizing Map (MLSOM) which aligns the visual elements in multiple label spaces. We use the proposed H_2OS feature and MLSOM method to answer these two questions for any query document: *when* was it written and *where*? The effectiveness and advantages of the proposed method will be demonstrated on our Medieval Paleographic Scale (MPS) data set [6]. A preliminary conference version of this work can be find in [19].

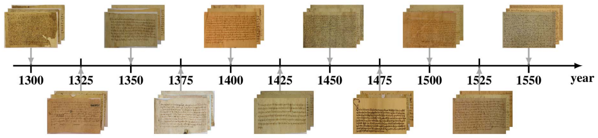


Fig. 2. The time line of the considered years. Each document is labeled with one of the 11 key year numbers and one of the four city names from 1300 to 1550 A.D.

TABLE I

THE NUMBER OF DOCUMENTS IN EACH KEY OF FOUR CITIES IN THE MPS DATA SET

City	1300	1325	1350	1375	1400	1425	1450	1475	1500	1525	1550	Sum
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

II. THE MPS DATA SET

This paper proposes a method for dating and localizing historical documents, using a reference data set consisting of documents from the late medieval period (1300-1550), and more precisely: charters. Charters, the most numerously available source type bequeathed to us from the Middle Ages, were public and formal declarations of legal or financial transactions or actions. Usually, the explicitly dated and chartered declaration was written on one side of a single piece of parchment.

The proposed data set consists of charters produced in four specific cities, namely Arnhem, Leiden, Leuven and Groningen. These four cities can be regarded as representing, more or less, different 'corners' of the Medieval Dutch language area. The charters were written mostly by professional scribes, keeping records in the cities. Often their writing careers covered about several decades. Hence, the evolution of writing styles as encountered in this data set is a very gradual process, and no obvious change would happen between two consecutive single years. Therefore, not all charters produced in our four cities in the period 1300-1550 are taken into account, but rather a set of charters assembled at chronological intervals. 'Key years' were chosen at every quarter century (1300,1325,1350,...,1550) (see Fig. 2) and only charters produced in these key years and within a period of five years before or after them were included, yielding a set of 2858 charters, grouped into 11 key years. Table I shows the numerical distribution of documents over the key years and the four cities. All documents in the MPS data set are labeled with a year and city of origin. More information can be found on the project website.[1]

III. HISTOGRAM OF ORIENTATIONS OF HANDWRITTEN STROKE DESCRIPTOR (H_2OS)

A. Motivation

The Histogram of Oriented Gradients (HOG) descriptor [17] is widely used to describe the mid-level visual elements [1], [3], [8], [20]. However, the HOG does not contain any scale information and is always used in a multi-scale strategy. Applying the HOG in handwritten document images is computationally inefficient because the resolution of the document images is always high (300dpi),

[1] http://application02.target.rug.nl/monk/Projects/MPS/

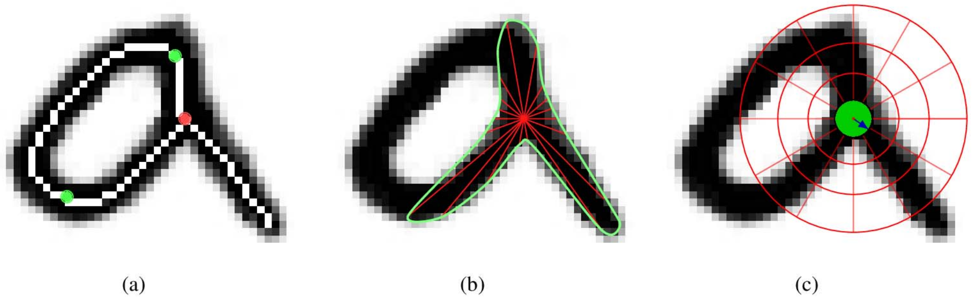


Fig. 3. Figure (a) shows the skeleton line detected in the handwritten character and the red point is the fork point and the green points are the high curvature points which are the candidate points for the H_2OS descriptor. Figure (b) shows the stroke width distribution around the fork point. The scale factor is determined by the minimum value of this stroke width distribution which reflects the stroke width on the fork point. Figure (c) shows the log-polar space with 3 rings and 12 orientations. The green circle in the center ring which is always filled by the ink pixels. The radius of the center ring (the blue line) is the scale factor of the proposed H_2OS descriptor.

leading to large image sizes. The SIFT [21] is a scale invariant feature and is also widely used when addressing handwritten documents. However, SIFT features directly extracted from the entire document images are not discriminative because the keypoints are located not only on the strokes but also on the background or near the contour of strokes [22], which introduces much noise. Usually, the SIFT detector is applied on the segmented word regions [23]. However, word segmentation is a challenging problem in images of historical handwriting. In addition, the computation of the SIFT features in these images with a high resolution is also far from efficient.

In order to solve these problems, we propose a novel descriptor named Histogram of Orientations of Handwritten Stroke Descriptor (HOHS or H_2OS for short), inspired by the Gradient Location-Orientation Histogram (GLOH) [24]. There are three main steps to build the H_2OS descriptor: (1) key-points selection; (2) scale-invariant log-polar space construction; (3) descriptor computation. Detailed information will be presented in the following sub-sections.

B. Key-Points Selection

We regard the structure points on the medial axis of handwritten strokes, such as the fork and high curvature points, as the key points. The structure points contain the topological information of the strokes and it has been shown in [25] that the regions around structure points contain discriminative information concerning writing styles. The procedure of the computation of structure points is as follows. First, the handwritten document is binarized and the medial axis (also known as skeleton line) is extracted by thinning methods. Choice of the binarization and thinning methods is problem specific and depends on the degradation of the historical document images. Common choices include the Otsu [26], Sauvola and Pietikäinen [27], and Moghaddam and Cheriet [28] for binarization and the method in [29] for thinning. Then the fork points are detected by [30] and the high curvature points are detected by the method in [25]. Fig. 3(a) shows an example of fork point and high curvature points detected on the skeleton lines.

C. Scale-Invariant Log-Polar Space Construction

The text in handwritten document images often has very inconsistent character sizes and document images are often digitized with different resolutions, which requires the use of a scale-invariant descriptor. In this paper, we consider the stroke width as the determinant for the scale factor because, usually one and the same or at least a similar writing instrument was used, (e.g., a quill), yielding a typical average stroke width. Given the key point p_i, the stroke width can be estimated using the method in [25] and [31] as follows. First, a stroke length distribution (see Fig. 3(b)) is built by computing the stroke length $len(\theta)$ from the key point p_i to the stroke boundary at each direction θ from 0 to 2π using the method proposed in [32]. Secondly, the stroke width $w_{stroke}(p_i)$ at the point p_i is estimated as the minimum value in the stroke length distribution:

$$w_{stroke}(p_i) \approx 2 \times \min_{\theta} len(\theta) \tag{1}$$

Given the key point p_i and the scale factor $w_{stroke}(p_i)$, a log-polar space can be built on p_i, which is the popular structure of the existed local descriptors, such as Shape Context [33], Self-Similar Descriptor (SSD) [34] and Histogram of Orientation Shape Context (HOOSC) [35]. The size of log-polar space is determined by the number of angular intervals N_{ang} and the number of distance intervals N_r. The distance intervals is equal to the half of the stroke width $w_{stroke}(p_i)/2$ in the log-polar space. An example of log-polar space is shown in Fig. 3(c). From the Fig. 3(c) we can see that the center ring in the log-polar space is always filled by the stroke ink and contain very little information. Therefore, we discard this region and, finally, there are $N_{ang} \times N_r$ bins in our H_2OS descriptor.

D. Descriptor Computation

For a given input handwritten image $\mathbf{I}$, we first compute the orientation map $\mathbf{G}_\theta$ on the orientation θ following [36] as:

$$\mathbf{G}_\theta = \left(\cos\theta \frac{\partial \mathbf{I}}{\partial x} + \sin\theta \frac{\partial \mathbf{I}}{\partial y}\right)^+ \tag{2}$$

where $(\cdot)^+$ is the operator such that $(a)^+ = \max(a, 0)$ to keep the only positive values to preserve the polarity of intensity changes [36], [37]. The gradient orientations in each region of the log-polar space are quantized in N_θ bins. In order to eliminate the quantized errors and avoid abrupt changes in the orientation map, a Gaussian kernel G_σ with the standard deviation σ is introduced to convolve the orientation map to obtain a smooth version:

$$\widetilde{\mathbf{G}}_\theta = G_\sigma * \mathbf{G}_\theta \tag{3}$$

Finally, the histogram of orientation in each region of the log-polar space on the point p_i is computed as:

$$\mathbf{h}^r = [\widetilde{\mathbf{G}}_1^r, \cdots, \widetilde{\mathbf{G}}_{N_\theta}^r] \tag{4}$$

where r denotes the index of region in the log-polar space, and $\mathbf{G}_i^r$ is the integrated value of the orientation map, $1 \leq i \leq N_\theta$, in the region r. The H_2OS descriptor is obtained by concatenating all $N_{ang} \times N_r$ histograms of orientation in the log-polar space, yielding a local feature vector with $N_{ang} \times N_r \times N_\theta$ dimensions. The descriptor is normalized dependent on each distance interval (or each ring) inspired by HOOSC.

E. Descriptor Analysis

The computation of the proposed H_2OS descriptor is very similar to SIFT [21] and HOG [17], and can be regarded as an extrapolation of GLOH [24] to handwritten document images. Therefore, it inherits all their properties. However, our proposed H_2OS descriptor is built on a scale-invariant log-polar space using the stroke width as the scale factor, thereby making the H_2OS scale invariant. In addition, the key points of our proposed H_2OS are always located inside the ink strokes, capturing the stroke structure information instead of the textural information of handwritten text.

The differences between our proposed H_2OS descriptor and the PSD descriptor [31] are that: (1) The PSD is a binarized-based descriptor while the proposed H_2OS is a gradient-based feature which contains more rich local information than the PSD descriptor. (2) The proposed H_2OS descriptor has a larger support region than the PSD, as show in Fig. 3, which could describe more context information around the key points. (3) The H_2OS descriptor captures the contrast and orientation information on the ink contours of handwritten images, which is more robust when dealing with the poor quality images.

The number of rings N_r of the H_2OS should be selected carefully, because if N_r is too small, the H_2OS will focus on local regions which contain little information, while if N_r is too large, the H_2OS will contain a lot of background noise. This can be observed in Fig. 4. We find that H_2OS almost always covers a meaningful visual element when $N_r = 3$, and this value is considered to build our H_2OS descriptor. We set the number of angular intervals of the log-polar space $N_{ang} = 12$ and the number of bins of the quantized orientations $N_\theta = 8$. Finally, the dimension of the H_2OS is $12 \times 3 \times 8 = 288$.

IV. Multi-Label Self-Organizing Map (MLSOM)

A. Motivation

Given the visual elements extracted by the proposed H_2OS feature, one possible way to discover the correlations between the visual elements and the labels, such as the year and the city, has been proposed in [1] and [3]. The general procedure is: (1) use the unsupervised cluster method (such as the k-means) to obtain the clusters in each label space; (2) select the discriminative clusters which correlate with their labels using an exhaustive-search method on the whole data set; and (3) train a SVM classifier for each selected cluster and find the correspondences across the entire data set. The main disadvantage of this approach is that it can not be directly used in multiple-label spaces at the same time.

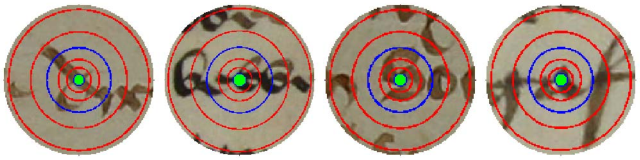


Fig. 4. Patches with different number of rings N_r (red circles). The patches with 3 rings (blue circle) cover the meaningful mid-level elements. Therefore, we set $N_r = 3$ to build the H_2OS descriptor.

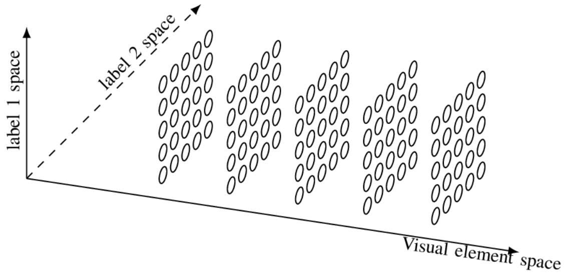


Fig. 5. An example of MLSOM with two label spaces and one visual element space in a 3D coordinate.

In this paper, we integrate these three steps into one framework, inspired by the property of the Self-Organizing Map (SOM) neural network which can preserve the topological properties of the input space using a neighborhood function. Moreover, we can integrate more than one label in the proposed framework and visual elements can align in multiple label spaces simultaneously.

B. MLSOM Configuration

In the traditional unsupervised SOM, the dimensional of the grid is usually low (1D or 2D). However, in the MLSOM we assume that each dimension corresponds to one label space. If there are L labels for each visual element, the dimension of the MLSOM will be $L + 1$, in which the extra dimension is the visual element space itself to preserve the topology of visual elements. Fig. 5 gives an example of MLSOM with 2 label spaces. Each node in the MLSOM neural network is connected to neighbor nodes in each label space (see Fig. 6).

C. MLSOM Training

Given the labeled training visual elements $\mathbf{v}_i = (\mathbf{x}_i, \mathcal{Y}^L)$ where $\mathbf{x}_i \in \mathcal{R}^d$ is the low-level representation of the i-th visual element, $\mathcal{Y}^L = \{y_i^1, y_i^2, \cdots, y_i^L\}$ is the label space and L is the number of labels, our aim is to align these visual elements

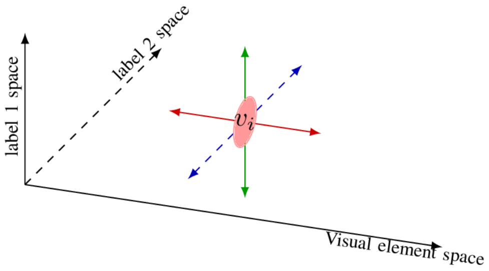


Fig. 6. An example of a visual element in multi-label space. The node is connected with neighbor nodes in each label space. (For example, the node v_i is connected in visual element space in the red line directions, connected in label 1 space in the green line directions and in label 2 space in the dashed blue line directions.) In this paper, we aim to align the visual element v_i among multi-label space simultaneously.

in the $L+1$ spaces (with the extra visual element space). In the traditional SOM model, there are two main training stages: the competitive stage and the cooperative stage. We adapt these two stages as follows to train our proposed MLSOM neural network.

Competitive stage: The basic idea of the competitive learning is that "only one cell or local group of cells at a time gives the active response to the current input [38]". In the competitive stage, the winner neuron is the one whose weight is most similar to the input vector, which is also known as the Best Matching Unit (BMU).

$$q^* = \arg\min_q \{\mathcal{D}(\mathbf{x}_i^{\mathcal{Y}^L}, \mathbf{w}_q^{\mathcal{Y}^L})\} \tag{5}$$

where $\mathbf{w}_q^{\mathcal{Y}^L}$ is the q-th neuron in the MLSOM with the same label as the training sample $\mathbf{x}_i^{\mathcal{Y}^L}$, q^* is the index of the winner neuron in the extra visual element space, $1 \le q^* \le \upsilon$ where υ is the dimension of visual element space and $\mathcal{D}$ is a distance function. The MLSOM has $L+1$ spaces with an extra visual element space, thus the searching is performed only on this extra visual element space, making sure that the BMU neuron has the same labels with the training sample. (Note that there is no label for the extra visual element space.)

Cooperative stage: Any neurons who are the neighbors of the BMU are updated their weights to preserve the topological order, by defining a neighborhood set N_{q^*}. The learning process is defined as:

$$\mathbf{w}_q(t+1) = \begin{cases} \mathbf{w}_q(t) + \eta(t)(\mathbf{w}_q(t) - \mathbf{x}_i) & \text{if } q \in N_{q^*}(t) \\ \mathbf{w}_q(t) & \text{if } q \notin N_{q^*}(t) \end{cases} \tag{6}$$

where $t = [0, T]$ is the epoch counter, T is the number of training iterations and the $\eta(t)$ is the learning rate which is defined, following [39], as:

$$\eta(t) = \left((\eta_T^{1/s} - \eta_0^{1/s})\frac{t}{T} + \eta_0^{1/s}\right)^s \tag{7}$$

where $s(> 0)$ is the steepness factor, η_0 and η_T are the starting and ending values. This is a decreasing function and the maximum value η_0 decreases to η_T.

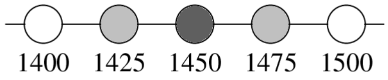


Fig. 7. An example of the key year label space (ordered label space) in MPS [6] with 25 years interval. Assume that the black neuron is the BMU, then the neighbors of the BMU are the neighbors in the coordinates of the MLSOM if the MLSOM network is initialized with the label order. For example, the neighbors of the BMU are the connected left and right neurons (the gray ones).

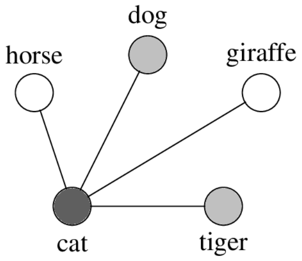


Fig. 8. An example of animal label space (non-ordered label space). Assume that the neurons represent the visual elements about **head** of animals and the black neuron is the BMU, then the neighbors of BMU should be determined by the similarity between the BMU and other neurons. For example, the top 2 neighbors of the head of the cat should be the heads of the dog and the tiger (according to the similarity of visual elements of the animal's heads). In fact, all the neurons have no semantic meanings and thus similarity is computed on the low-level feature space.

The most important part of MLSOM is the definition of the neighborhood set N_{q^*}. Traditionally, a Gaussian function is widely adopted as the neighbor function in the unsupervised SOM, which finds the coordinate neighbors around the BMU in the SOM neural network. However, in the proposed MLSOM, we want to discover the connections in the label spaces and the coordinate neighbors are not always the neighbors in the label space.

There are two types of visual element label space: the ordered label space and non-ordered label space. In the ordered label space, the labels have an inherent order. For example, y_{i+1} always follows y_i. There are many different ordered label spaces, such as a time sequence, the year (key year) labels and people's ages. Fig. 7 shows an example of a key year label space, which is an ordered label space. If we initialize the MLSOM with the same order as labels, the neighbors of the BMU in the label space is the same as the neighbors of the BMU in the coordinates of the MLSOM neural network, which can be computed as:

$$N_{q^*}^{y_i}(t) = \left[q^{*y_i} - r(t), q^{*y_i} + r(t)\right] \tag{8}$$

here y_i is the index of label space and $r(t)$ is the spatial resolution of the neighbors at epoch t, which can be determined by Eq. 7. Note that $r(t)$ is the decreasing function, which indicates that at the beginning of the training, the SOM aims to build a general connection in a large scope in the y_i label space and at the end of the training $r(t)$ tends to zero and the MLSOM is finely tuned to preserve the discriminative information.

In the real-world, most labels are non-ordered, such as the categories of animals or cities (see an example in Fig. 8). In this case, the neighbors of the BMU in the coordinate of the MLSOM neural network can not reflect the neighbors in the label space and the Eq. 8 can not be used as the neighbor function. In this paper, we assume that the BMU neuron is fully connected to other neurons in the non-ordered label space and the neighbors of the BMU are determined as the top $N_{top}(t)$ similar neurons according to the distance function $\mathcal{D}$. The $N_{top}(t)$ is the spatial resolution in this non-ordered label space and can also be determined using Eq. 7. Fig. 8 shows an example of the neighbors of the BMU in the non-ordered space.

D. Learning From Neighbors

Zero-shot learning aims to learn the labels of an image, in the case where no visual examples of that labels are available during training [40]. The relationships between the annotated classes and the unseen classes are built usually on a high level, such as the attributes [41] or co-occurences of visual concepts [40]. This information is often learned from a large labeled data set. The assumption of this paper is that the visual elements of a certain class have a subtle but consistent difference with its neighbors, which can be captured by the proposed MLSOM neural network. Therefore, our method can also be used for zero-shot learning by learning from the neighbors. In the earlier training stage, the neurons in the MLSOM are updated not only by the training visual elements when they are the winner neurons, but also by the visual elements when their neighbor neurons are the winner neurons. Therefore, each neuron in the MLSOM also learns from their neighbors until the window size closed to zero. If there is no training sample in a certain label space, the corresponding neurons will be updated by the neighbors and thus contain the average information of their neighbor neurons.

E. MLSOM Voting

Each neuron in the trained MLSOM neural network carries labels. Therefore, the trained MLSOM can be directly used to predict the labels of the test images. The visual elements $\mathbf{v}_i$ from the test image $\mathbf{I}$ can be mapped into the learned MLSOM space to form a histogram h^{L+1}, which is similar with the bag-of-word model [42]. The probability $p(y_i^l)$ that the test image $\mathbf{I}$ belongs to the label y_i^l in l-th label space is estimated by:

$$p(y_i^l) = \frac{\sum_{q=0,q\neq l}^{L} h^q(i)}{\sum_j \sum_{q=0,q\neq l}^{L} h^q(i)} \tag{9}$$

here $\sum_{q=0,q\neq l}^{L} h^q(i)$ counts the number of occurrences of corresponding visual word in MLSOM along other label spaces except l-th label space and the $\sum_j \sum_{q=0,q\neq l}^{L} h^q(i)$ is the total number of occurrences. Finally, the test image is assigned to the label with the highest probability:

$$y^l(\mathbf{I}) = \arg\max_{y_i} \{p(y_i^l)\} \tag{10}$$

V. Experiments

A. Document Stroke-Shape Elements

Generally, the stroke structures are very often repeated in handwritten texts of historical documents (Fig. 9), because the number of letters in an alphabet is limited and their appearances in the feature space are quite similar due to style considerations in the writer. Therefore, regarding them as separate visual elements makes inefficient use of the available information. Our goal in this section is to detect a set of primary visual elements that represents the wide variety of stroke structures in each historical document, which contains the discriminative information concerning writing style. We call the detected primary visual elements **Stroke-Shape Elements**. We assume that each Stroke-Shape Element represents a style element. The stroke-shape elements are clusters of these sampled patches generated by the k-means, algorithm from the H$_2$OS descriptor using the χ^2 distance. Typical individual cluster centroids are illustrated in Fig. 10. The main reason we use the Stroke-Shape Element instead of the sampled patch instances in the document is to avoid the effects of an unbalanced number of instances for each Stroke-Shape Element in the voting stage. For example, a large number of instances of the character ‘e’ with the same writing style will lead to a large number in the corresponding bin in the voting histogram. However, if we use the Stroke-Shape Element representation, there is only one writing style element contributing to the voting histogram.

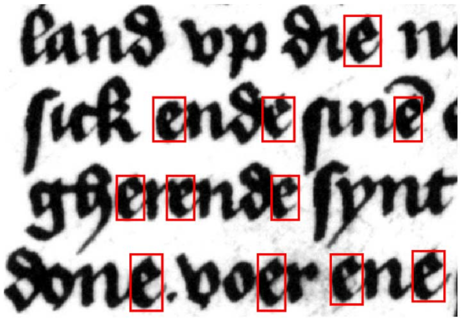


Fig. 9. An example of repeated visual elements in a part of a historical document.

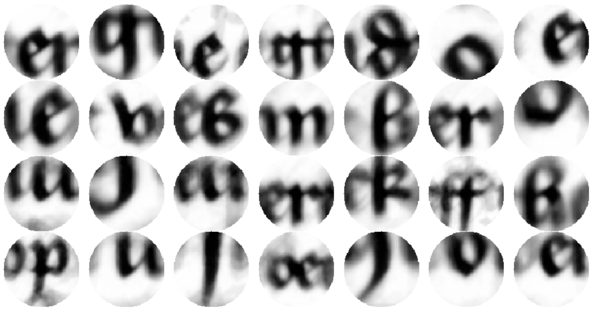


Fig. 10. Examples of the Stroke Shape Elements learned from a document image (part of it is shown in Fig. 9). These Stroke Shape Elements are the primary visual elements in the given historical document which contain the discriminative information of the writing style.

B. Experimental Setup

There are two label spaces: The key-year space and the city space and one extra visual-element space for historical document dating and localization. The key-year space is an ordered label space and the size is 11, while the city space is a non-ordered label space and the size is 4. The size of the extra visual element υ should be set manually. The parameters in the training of the MLSOM neural network are set as follows: the starting value of the learning rate $\eta(t)$ is set $\eta_0 = 0.5$ and the end is set $\eta_T = 0.005$; for the spatial resolution of the neighbors $r(t)$, the starting value is set $r(0) = 5$ and the end is set $r(T) = 0$. A steepness factor of $s = 5$ was used to compute the learning rate and neighbor size, following [39] and the number of training iterations is $T = 500$.

When addressing the dating problem, the Mean Absolute Error (MAE) and Cumulative Score (CS) are often used to measure the performance of the system. The MAE is a Manhattan-type distance, which is typically defined as:

$$MAE = \sum_{i=1}^{N} |\overline{K(y_i)} - K(y_i)|/N \tag{11}$$

where $K(y_i)$ is the ground-truth of the input document y_i and $\overline{K(y_i)}$ is the estimated key year, N is the number of test documents. The Cumulative Score(CS) is typically defined as [11]:

$$CS(\alpha) = N_{e\leq\alpha}/N \times 100\% \tag{12}$$

where $N_{e\leq\alpha}$ 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.

In the MPS data set, some documents have writer labels and the writers of the rest of the documents are unknown. In order to maintain the high variance and capture the general writing styles during the training, we randomly select only one document from each writer and also randomly select one document from those whose writers are unknown. In addition, in order to avoid effectively practicing writer identification when dating, we randomly split the documents with writer labels into four parts to generate four training sets. Finally, five different subsets are used to train the MLSOM neural network and the average performance is reported in this paper.

C. MLSOM Voting Results

There are two parameters that need to be optimized: the number of clusters k for the stroke shape elements and the size of visual elements spaces υ. We used a grid search method to find the appropriate values and Table II shows the performances of historical document dating with different values. From Table II we can see that increasing the number of stroke shape elements k does not improve the performance, which results from the fact that the number of different writing styles is limited in one document and a large value of k introduces more noise. The best value of k is 100 in our experiments for different values of υ. Conversely, the performance is better when the value υ is higher. However, a high value of υ also needs a large memory and long computing time. In our experiments, we set the value of υ to 300, which achieves the best performance (the MAE is 25.9 years). The localization precision is shown in Table III. The performance is higher when k is smaller and υ is higher, which is the same as the dating performance in Table II. The best performance for localization is achieved when $k = 50$ and $\upsilon = 400$ and 83.8% documents are correctly localized.

TABLE II

THE AVERAGE AND STANDARD DEVIATION MAEs FOR DATING WITH DIFFERENT VALUES OF THE NUMBER OF CLUSTERS k AND THE SIZE OF VISUAL ELEMENTS SPACES υ

		υ				
System		50	100	200	300	400
k	50	35.2±1.6	30.9±2.6	28.2±3.6	28.0±3.8	27.8±3.9
	100	35.4±2.0	30.4±2.7	27.3±3.4	**25.9**±4.0	**25.9**±4.4
	200	37.3±3.3	33.1±3.4	28.5±3.1	27.1±3.2	26.2±4.2
	300	38.1±3.7	33.6±2.9	29.0±3.5	27.5±3.6	26.8±4.2
	400	39.8±3.6	33.4±2.5	29.7±2.6	28.1±3.6	26.9±4.1

TABLE III

THE AVERAGE AND STANDARD DEVIATION PRECISIONS (%) FOR LOCALIZATION WITH DIFFERENT VALUES OF THE NUMBER OF CLUSTERS k AND THE SIZE OF VISUAL ELEMENTS SPACES υ

		υ				
System		50	100	200	300	400
k	50	77.0%±3.6	80.2±4.8	82.4±4.0	83.2±3.4	**83.8**±4.1
	100	75.8%±4.5	78.7±4.5	80.6±4.8	81.8±4.7	82.8±5.1
	200	76.8%±4.2	78.0±5.2	81.3±5.8	81.3±5.8	81.8±4.4
	300	75.1%±3.1	74.9±6.2	78.9±5.6	81.1±5.3	80.8±5.3
	400	75.0%±4.4	75.9±4.5	78.1±6.2	79.4±5.3	81.4±4.7

TABLE IV

THE MAEs, STANDARD DEVIATIONS (STD) AND CSs($\alpha = 25$) ON OUR DATABASE WITH DIFFERENT METHODS

Method	MAE	CS(α=25)
Random Guess	85.3±58.5	25.7%
Monk [9]	36.0±20.6	-
Study[6]	35.4	63.5%
MLSOM voting	25.9±4.5	73.7%

Table IV shows the MAEs of the proposed method compared with our previous methods, as well as a random guess. The method which is from the Monk system [9], [43] used the human labeled characters for dating and the work in [6] used global and local regression methods with the Hinge and Fraglets [44] features. Our method improves the performance and the best MAE is 25.9 years which is almost 10 years lower than previous results. Fig. 11 shows the CS measures for different methods. We can observe that the proposed method also improves the score for lower error level, e.g., $\alpha \leq 25$ years. For example, 46.2%(±7.2) documents are estimated correctly by the proposed method, which is higher than the 22.5% given by the method in [6]. When the year error level is 25 years, the proposed method could improve the accuracy by 10.2% (see the last column in Table IV).

Fig. 12 provides the results of the localization precisions with the year error levels from 0 to 100 years. 41.5%(±6.3%)

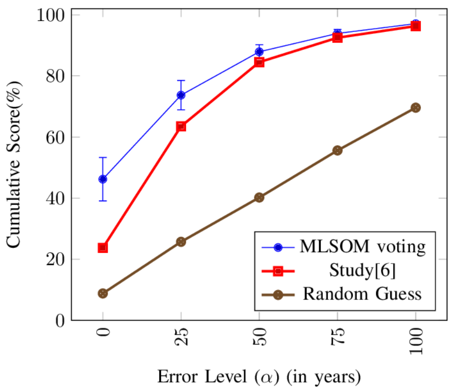


Fig. 11. Cumulative Scores of $p(MAE \leq \alpha)$ on the error levels (α) from 0 to 100 years.

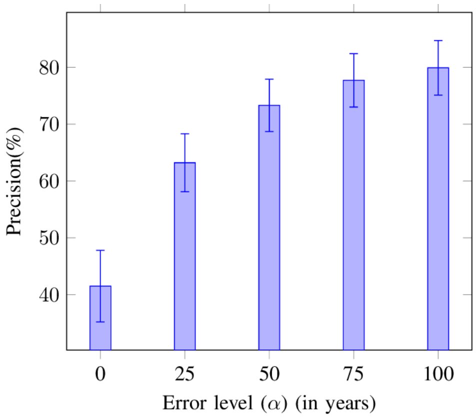


Fig. 12. The localization precision with different year error levels from 0 to 100 years.

TABLE V

THE ACCURACY (%) OF THE LOCALIZATION CONFUSION MATRIX AMONG THE FOUR CITIES

City	Arnhem	Leiden	Leuven	Groningen
Arnhem	86.2±3.0	7.6±3.3	2.7±1.2	3.5±1.2
Leiden	18.1±5.6	68.4±9.2	10.2±6.7	3.3±0.5
Leuven	0.4±0.6	2.2±2.8	97.0±3.5	0.4±0.5
Groningen	4.6±3.9	2.1±1.8	1.3±0.6	92.0±4.5

documents are estimated correctly with the date and local information. When the year error level is 25 years, the precision of the localization is 63.2%(±5.1%). The results demonstrate that dating and localizing simultaneously is more difficult than dating and localizing separately.

Table V shows the confusion matrix for the historical document localization. From the table we can observe that the documents from Leuven and Groningen are quite easy to localize than those from Arnhem and Leiden. The precision for Leiden is only 68.4%(±9.2%) and the documents from Leiden are easy to be estimated as documents from Arnhem and Leuven.

D. Dating by Classification

The dating problem can be considered as a classification problem because the document distribution in the considered period has a obvious borders between the nearby key years in the MPS data set according to the domain experts (paleographers). We assume that all the documents from the same key year form a class and there are 11 classes (11 key years) in our MPS data set. We train 11 classifiers using a linear SVM with the one-versus-all strategy. The parameter C is estimated by a grid search method in the range of $\{2^{-16}, 2^{-15}, \cdots, 2^{15}, 2^{16}\}$. The query document is assigned to the key year with the maximal SVM score among the 11 trained classifiers.

The proposed MLSOM is also a cluster method which can be used to train the codebook. We map the extracted patches described by H2OS features to the codebook to compute the representation of the documents using the bag-of-word model, which is denoted as $\text{H}_2\text{OS}_{mlsom}$. We believe that the codebook trained by the MLSOM is more discriminative than the traditional SOM methods because it contains the label information. In order to evaluate this observation, we train a codebook using the traditional SOM method with the same data and the same parameters, which is denoted as $\text{H}_2\text{OS}_{som}$.

We also compare our method with the existing features used for writer identification in handwritten documents, which can be typically divided into two categories: textural-based and grapheme-based features. Two textural-based, such as Hinge and Quill, and two grapheme-based features, such as Junclets and Strokelets, are selected in the experiments, which are:

Hinge: The Hinge is a texture level feature [39], [44], which captures the slant and curvature information of the handwritten ink trace. The Hinge feature is a joint probability distribution of the orientations of the two edge fragments constituting the legs of an imaginary hinge. The two parameters of the Hinge feature, the number of angle bins p and th leg length q, are set to $p = 40$ and $q = 20$.

Quill: The quill feature was designed to capture the property of the capillary-action of writing instruments, such as the "quill pen" which were used until the 19th century [32]. The Quill feature is a probability distribution of the relation between the ink direction and the ink width. The parameters are set following the original paper [32].

Junclets: The junction regions are important visual elements in handwritten documents which reflecting the writing style. The junction feature proposed in [25] computes the distribution of the stroke length on the junction point in a polar space. The Junclets is the probability-density function of the junctions based on a trained codebook with the size of 625.

Strokelets: The connected components of the handwritten text are segmented into sub-strokes on the fork points and the Polar Stroke Descriptor which is similar with the junction feature is used to describe the sub-strokes [31]. The Strokelets is also a probability-density function of the sub-strokes based on a trained codebook with the same size of Junclets.

After computing the feature representations of documents in the MPS data set, we perform dating in two ways: Dating while excluding writer duplicates versus inclusion of writer duplicates. The exclusion of writer duplicates enforces a style-based dating, as opposed to a dating result which can be attributed to writer identification. In the following sections, we compare the performance of the proposed $\text{H}_2\text{OS}_{som}$ and $\text{H}_2\text{OS}_{mlsom}$ representations, as well as the Hinge, Quill, Junclets and Strokelets features in the same experimental setting.

TABLE VI

THE DATING PERFORMANCE WITH DIFFERENT CONFIGURATIONS WHEN EXCLUDING WRITER DUPLICATES

Feature	MAEs	CS(α=25)
Hinge [44]	22.1±2.9	80.6%±3.1
Quill [32]	23.7±2.9	80.5%±3.1
Junclets [25]	21.7±3.7	79.4%±4.4
Strokelets [31]	19.4±2.7	83.1%±2.6
H_2OS_{som}	25.2±3.2	76.4%±3.7
H_2OS_{mlsom}	**15.9**±2.9	**85.4**%±3.7

TABLE VII

THE PERFORMANCE OF THE DATING WITH DIFFERENT FEATURES WHEN INCLUDING WRITER DUPLICATES

Feature	MAEs	CS(α=25)
Hinge [44]	12.2±0.9	89.6%±1.3
Quill [32]	12.1±1.0	89.5%±1.3
Junclets [25]	12.4±0.7	88.4%±1.4
Strokelets [31]	11.4±0.9	89.4%±1.7
H_2OS_{som}	15.8±1.8	85.4%±1.6
H_2OS_{mlsom}	**9.1**±0.8	**91.4**%±1.5

1) Dating Results When Excluding Writer Duplicates: In the MPS data set, the writers of several documents are known and the writers of the rest of the documents are unknown. In order to avoid effectively practicing writer identification when dating, we carefully and randomly split the data set into training(70%) and testing(30%) sets to make sure that the same writer never appears in both the training and the testing set, which means that all documents produced by the same hand should be only in the training set or only in the testing set. The experiment is repeated 20 times and the average results with the standard deviation are reported.

Table VII shows the MAEs of different methods. We can see that our proposed H_2OS_{mlsom} achieves the best performance in terms of both MAE and CS(α=25). The performance of the H_2OS_{mlsom} method using the codebook trained by the proposed MLSOM method is much better than the H_2OS_{som} which uses the traditional cluster method to train the codebook. The H_2OS_{mlsom} reduces the MAE by 9.3 years, which demonstrates that the proposed MLSOM codebook contains more discriminative information than the codebook trained by traditional cluster methods.

The performance of the proposed H_2OS_{mlsom} outperforms other methods. In the existing features, the Strokelets achieves the best performance. However, the MAE of the Strokelets is 19.4 years, which is lower than the proposed H_2OS_{mlsom} by 3.5 years.

The cumulative score of the different methods is shown in Fig. 13, from which we can see that the proposed H_2OS_{mlsom} method outperforms all other methods, especially when the error level is equal to zero. 66.8% documents are estimated correctly by the H_2OS_{mlsom} method, while only 53.8%, 50.2%, 52.4%, 55.2%, 48.2% documents are estimated correctly by the Hinge, Quill, Junclets, Strokelets and H_2OS_{som}, respectively.

2) Dating Results When Including Writer Duplicates: In this section, we conduct the experiment of the dating on the MPS data set by randomly splitting the documents into training(70%) and testing(30%) sets without considering whether the documents from the same hand appear in the training set or in the testing set. Table VII shows the MAEs and the CS with 25 error level for different methods. When using the proposed H_2OS feature, the MAE of the H_2OS_{mlsom} is higher than the MAE of the H_2OS_{som} by 6.7 years. It also improves the measure CS(α=25) from 85.4% to 91.4%, which means 91.4% documents are correctly estimated with the error equal or less than 25 years by the H_2OS_{mlsom}. Compared to other features, our proposed H_2OS_{mlsom} achieves the best performance in terms of MAEs (9.1 years) and CS(α=25) (91.4%). The results are very good, however, entailing a contamination over writer identity which may, or may not bother the end user.

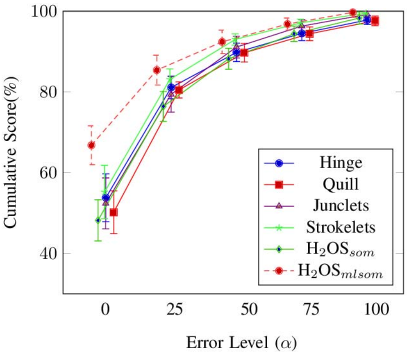


Fig. 13. Comparisons between our proposed methods and other methods in terms of the cumulative scores when excluding writer duplicates. H_2OS_{mlsom} results are a clear improvement over other studies.

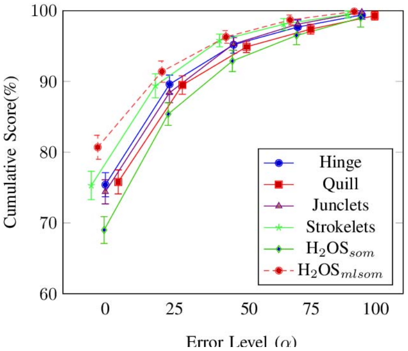


Fig. 14. Comparisons between our proposed methods and other methods in terms of the cumulative scores when including writer duplicates.

Fig. 14 shows the cumulative score comparisons among different methods. Our proposed H_2OS_{mlsom} performs much better than other methods, especially when the error level is less than 50 years. The CS(α=0) of the H_2OS_{mlsom} method is higher than 80%, while the CS(α=0) of other methods are lower than 80%. The MAEs and the number of documents in each key are shown in Fig. 15. From this figure we can infer that the performance in each key year has a relationship with the number of documents. For example, the number of documents in 1450 is the largest, leading to lowest performance of all methods among the 11 key years. Table VIII shows the MAEs of three different periods: 1300-1375, 1400-1475 and 1500-1550. From the table we can see that dating documents

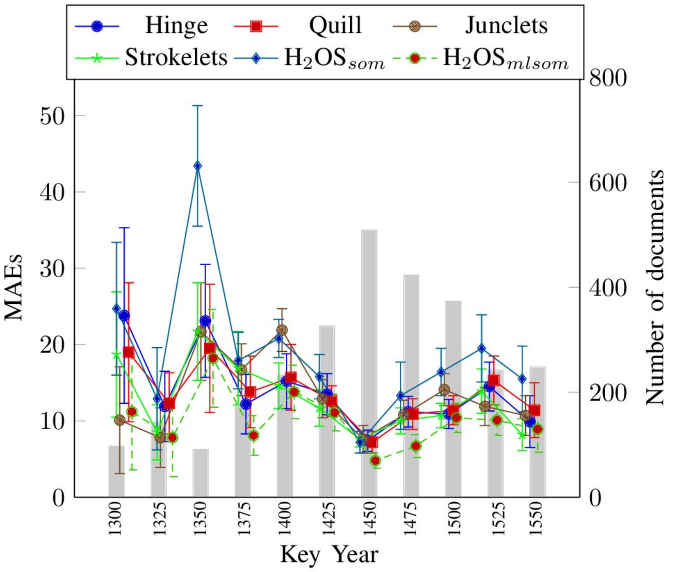


Fig. 15. The average MAEs and the number of documents in each key year.

TABLE VIII
THE AVERAGE MAEs AND THE NUMBER OF DOCUMENTS IN THREE PERIODS

Feature	1300-1375	1400-1475	1500-1550
Hinge [44]	17.7± 6.9	11.8±2.4	8.8±2.1
Quill [32]	16.2±6.6	11.6±2.4	9.5±2.2
Junclets [25]	15.0±4.9	13.1±2.3	9.3±2.0
Strokelets [31]	16.5±5.4	10.9±2.3	7.9±1.8
H_2OS_{som}	22.7±6.2	11.4±1.9	10.9±1.9
H_2OS_{mlsom}	11.6±4.2	7.2±1.4	6.7±1.5
Number of documents	505	1490	873

TABLE IX
THE MAEs PERFORMANCE WITH CODEBOOKS TRAINED WITH DIFFERENT CONFIGURATIONS CONSIDERING THE WRITER-RELATED BIAS

Feature	Codebook *wr.incl.*	Codebook *wr.excl.*
H_2OS_{som}	15.8	17.2
H_2OS_{mlsom}	9.1	13.5

from the period of 1500-1550 is much easier than documents from the other two periods.

3) Stability of Codebooks Trained With Writer-Related Bias: In this section, we evaluate the performance of dating with the different MLSOM codebooks trained by documents from a subset of writers in each key year in order to evaluate the performance with writer-related bias. Table IX shows the performance with codebook trained by documents from all writers (denoted as Codebook *wr.incl.*) and with codebook trained by documents only from a subset of writers (only one-fourth writers are involved in this experiment, denoted as Codebook *wr.excl.*). From the table we can see that including all writers in the codebook training provides slightly better results. This shows that handwritten patterns written by different writers in the MPS data set are variable and it is better to train the codebook with all writers.

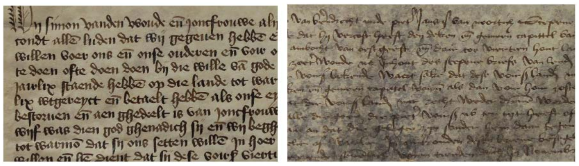


Fig. 16. Examples of documents in the MPS data set with high quality (left figure) and low quality (right figure).

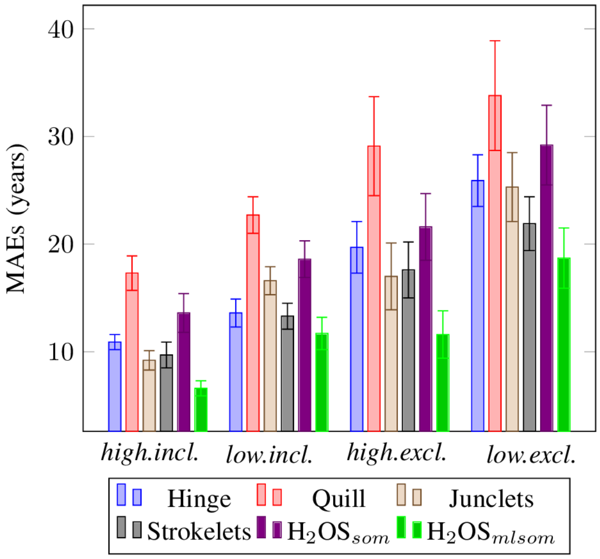


Fig. 17. The MAEs of different features with different image quality. *high.incl.* and *high.incl.* mean the performance with high quality when including writer duplicates (*incl.*) and excluding writer duplicates (*excl.*), respectively. *low.incl.* and *low.incl.* mean the performance with low quality.

4) Dating Results With Different Image Quality: Some documents in the MPS data are heavily degraded (low quality, see Fig. 16). In order to evaluate the performance with different image quality, we split all documents in MPS into high-quality and low-quality sets. We manually select 109 documents with very high quality and 118 documents with very low quality to train a kernel SVM to predict the quality of all documents on the MPS data set. Seven features are used to represent the documents as follows: f1: the absolute distance between the mean values of the Gaussian distributions of the ink and background pixels; f2: the ratio between the standard deviations of these two Gaussian distributions; f3: the distribution of the number of connected components; f4: the entropy of the distribution of the contour length of connected components; f5: the entropy of the distribution of the stroke width estimated by the method [32]; f6: the ratio of the uniform LBP pattern and non-uniform ones [45]; f7: the entropy of the distribution of the line length computed by the fast line detector (LSD) [46]. In practice, we have found that these seven features work very well and the accuracy is over 90% on the MPS data set.

Fig. 17 shows the performance with different image qualities. From the figure we can see that (1) the performance of all features shows the same trend as the one without considering the image quality: Quill < H_2OS_{som} < Hinge < Junclets

TABLE X
THE MAEs PERFORMANCE OF THE DATING WITH DIFFERENT QUALITY CONFIGURATIONS

Feature	Train: high Test: low	Train: low Test: high	Difference
Hinge [44]	26.8	15.7	11.1
Quill [32]	33.6	26.1	7.5
Junclets [25]	23.6	13.6	10.0
Strokelets [31]	21.9	16.4	5.5
H_2OS_{som}	26.0	26.5	0.5
H_2OS_{mlsom}	18.3	15.6	2.7

TABLE XI
THE DATING PERFORMANCE WITH DIFFERENT METHODS

	Excluding duplicates		Including duplicates	
Feature	MAEs	CS(α=25)	MAEs	CS(α=25)
LBP [45]	39.9±4.9	59.3%±6.3	34.3±8.1	65.2%±7.7
SIFT [21]	33.9±4.3	69.9%±3.7	23.3±1.2	78.2%±1.1
Quill [32]	23.7±2.9	80.5%±3.1	12.1±1.0	89.5%±1.3
Hinge [44]	22.1±1.9	80.6%±3.1	12.2±0.9	89.6%±1.3
Junclets [25]	21.7±3.7	79.4%±4.4	12.4±0.7	88.4%±1.4
CO^3 [39]	20.3±2.9	82.1%±3.2	11.5±0.8	89.5%±1.5
Strokelets [31]	19.4±2.7	83.1%±2.6	11.4±0.9	89.4%±1.7
kCF [16]	19.2±3.5	85.8%±2.8	10.8±0.9	90.8%±1.1
kSF [16]	17.4±1.9	86.8%±2.0	9.9±0.6	91.8%±1.5
H_2OS_{som}	25.2±3.2	76.4%±3.7	15.8±1.8	85.4%±1.6
H_2OS_{mlsom}	15.9±2.9	85.4%±3.7	9.1±0.8	91.4%±1.5

< Strokelets < H_2OS_{mlsom} according to their performance and (2) the results on the high-quality set are better than the results on the low-quality set for all features.

We also conduct the experiment of dating by using a high-quality set for training and a low-quality set for testing, and vice versa. Table X shows the MAEs of different features and we can see that the results of using low-quality set for training are better than using high-quality set. One important observation that can be made is that our proposed H_2OS_{som} and H_2OS_{mlsom} methods are much more stable when dealing with image qualities, achieving the minimum differences between the MAEs when using different image qualities for training and testing.

5) Comparison With Other Studies: In this section, we conduct experiments of dating using more features, such as the SIFT [21] extracted based on the coarse word zones, [45] (with 255 patterns without the background), CO^3 [39], kCF (combined with k=2,3,4,5) and kSF (combined with k=1,2,3) [16]. Table XI shows the MAEs of different features. From the table we can see that our proposed method achieves better results than other features. In addition, the H_2OS feature is very efficient to represent the handwritten visual elements, such as the Stroke Shape Elements shown in Fig. 10.

E. Geographical Localization by Classification

To evaluate the document localization performance, we train four classifiers for the four cities. Fig. 19 and Fig. 18 show the precision of the four cities including the writer duplicates and excluding the writer duplicates, respectively. From the two figures we can find that our proposed methods outperform all the other methods and the H_2OS_{mlsom} achieves the best performance in the two configurations. The precision of the H_2OS_{mlsom} for Leuven is 90.5% when including writer duplicates and is 86.5% when excluding writer duplicates, which are explicitly higher than other methods. Table XII gives the confusion matrix of the localization of the proposed H_2OS_{mlsom} and the average precision for geographical localization is 93.8%.

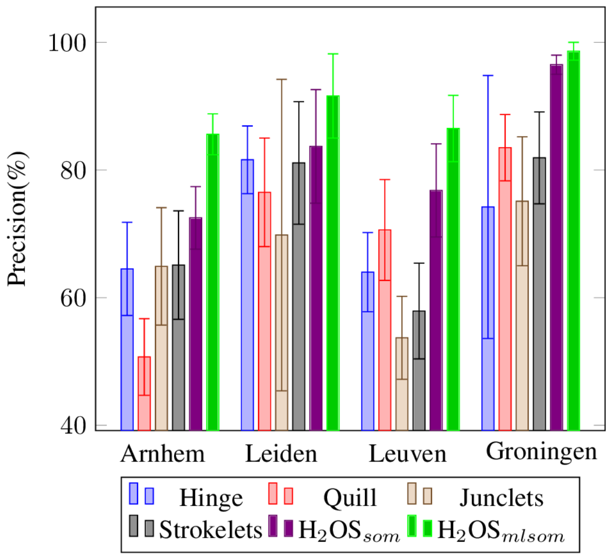


Fig. 18. The localization precision with different methods in the four cities when excluding writer duplicates.

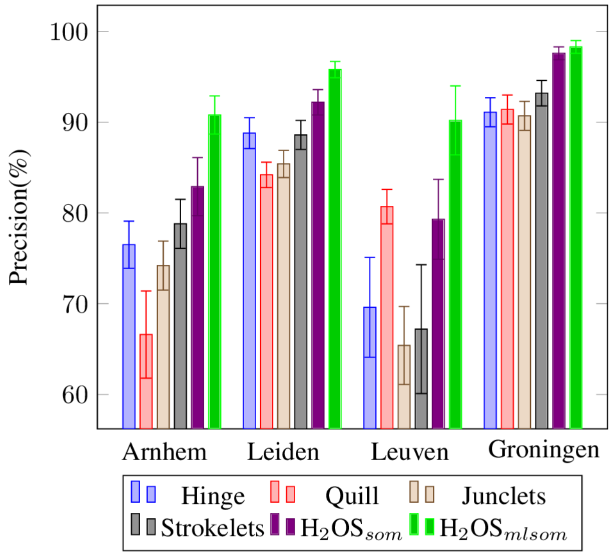


Fig. 19. The localization precision with different methods in the four cities when including writer duplicates.

F. Results of Learning From Neighboring Neurons

In this section, we evaluate the performance of the learning from neighbors for dating and localization excluding training samples from the target year and city. In this condition, the intrinsic interpolation by the Kohonen map should allow for estimating the target year and city. We leave each combination

TABLE XII

THE ACCURACY (%) OF THE LOCALIZATION CONFUSION MATRIX AMONG THE FOUR CITIES USING THE H_2OS_{mlsom} METHOD WHEN INCLUDING WRITER DUPLICATES

City	Arnhem	Leiden	Leuven	Groningen
Arnhem	90.9±2.1	7.0±2.0	0.5±0.6	1.6±1.1
Leiden	2.8±0.9	95.8±0.9	0.6±0.4	0.8±0.4
Leuven	0.1±0.4	8.9±3.7	90.2±3.8	0.8±0.9
Groningen	0.6±0.4	0.8±0.6	0.3±0.3	98.3±0.7

TABLE XIII

THE MAEs AND CSs($\alpha = 25$) WHEN LEARNING FROM NEIGHBORING YEARS AND CITIES EXCLUDING ALL SAMPLES FROM THE TARGET YEAR AND CITY FROM THE TRAINING SET

Method	MAE	CS(α=25)
Random Guess	85.3±58.5	25.7%
Monk	36.0±20.6	-
Study[6]	35.4	63.5%
Learning from neighbors	38.9	63.5%

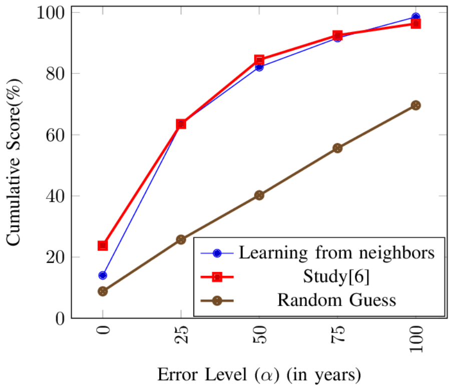


Fig. 20. Cumulative Scores of $p(MAE \leq \alpha)$ on the error levels (α) from 0 to 100 years when learning from neighboring years and cities excluding all samples from the target year and city from the training set.

of labels (year and city) out and train the MLSOM using the rest documents. After that, the documents with the leave-out labels are used to test the trained MLSOM neural network. Table XIII shows the MAEs of the results compared to other studies and the CS score is shown in Fig. 20. From the results we can observe that due to this local mutilation of the Kohonen map, the MAE approximately equal to 39 years. This is quite natural because the corresponding neurons of the missed labels now contain the general or average information of their neighbors. Although missing the target year and city training data, the MLSOM can still learn the information from their neighbors and 14.0% documents are correctly dated (see Fig. 20 when $\alpha = 0$). For error level α=25 and higher, the CS scores are almost the same as the method in [6]. For historical document localization, 70% documents are localized correctly, compared to 83.8%, the best performance in Table III.

G. Discussion

The proposed MLSOM neural network contains the year and city label information and can be directly used for predicting the labels of the query document by voting. The results of voting method outperforms the existing systems. If no training data set is available for certain labels, the corresponding neurons learn the average information from their neighbors, with some degradation, but still comparable to our previous results.

In addition, our proposed MLSOM can be considered as a cluster method which contains more discriminative information than traditional cluster methods. The unsupervised cluster methods (such as regular SOM or k-means) discard the subtle difference among labels and are less discriminative in contrast to the proposed MLSOM method. The performance of dating and localization on the MPS data set based on representations using the MLSOM with the classification method achieves state-of-the-art results.

VI. CONCLUSION

In this paper, we studied the problem of historical document dating and localization using our new MPS data set. In order to extract the visual elements in historical documents, we developed the H_2OS feature which is a scale-invariant descriptor. We then proposed the Multiple-Label guided MLSOM method to align the visual elements in multiple label space. Our proposed MLSOM can be used for predicting labels directly, or can be used as zero-shot learning by learning from neighbors or can be used as a cluster method. The experimental results in relation to the MPS data set clearly show the efficacy of the proposed method. The best MAE on the MPS data set was 15.9 years when excluding writer duplicates and 9.1 years when keeping writer duplicates in the reference set.

Future work includes: (1) applying our proposed H_2OS feature to other types of document analysis, such as writer identification and word spotting. (2) applying our proposed MLSOM method to other visual recognition applications, for example, visual element alignment across the same object in different images and dating color images. (3) using other learning methods for dating and localization, such as the attribute learning and deep learning methods.

REFERENCES

[1] Y. Lee, A. Efros, and M. Hebert, "Style-aware mid-level representation for discovering visual connections in space and time," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2013, pp. 1857–1864.

[2] [Online]. Available: https://nl.wikipedia.org/wiki/Google_Street_View

[3] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes Paris look like Paris?" *ACM Trans. Graph.*, vol. 31, no. 4, pp. 103–110, Dec. 2012.

[4] L. Shijian and C. L. Tan, "Script and language identification in noisy and degraded document images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 1, pp. 14–24, Jan. 2008.

[5] D. Ghosh, T. Dube, and A. P. Shivaprasad, "Script recognition—A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2142–2161, Dec. 2010.

[6] S. He, P. Sammara, J. Burgers, and L. Schomaker, "Towards style-based dating of historical documents," in *Proc. Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Sep. 2014, pp. 265–270.

[7] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1178–1188, Jul. 2008.

[8] S. Singh, A. Gupta, and A. A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *Proc. Int. Conf. Comput. Vis. (ECCV)*, Berlin, Germany, 2012, pp. 73–86.

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

[10] P. Samara, "Towards a medieval palaeographical scale," in *Papsturkundenforschung Zwischen Internationaler Vernetzung und Digitalisierung*. 2014.

[11] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, Dec. 2007.

[12] K.-Y. Chang and C.-S. Chen, "A learning framework for age rank estimation based on face images with scattering transform," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 785–798, Mar. 2015.

[13] F. Palermo, J. Hays, and A. A. Efros, "Dating historical color images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 499–512.

[14] F. Wahlberg, L. Mårtensson, and A. Brun, "Large scale style based dating of medieval manuscripts," in *Proc. Workshop Historical Document Image Process. (HIP)*, 2015, pp. 107–114.

[15] N. R. Howe, A. Yang, and M. Penn, "A character style library for Syriac manuscripts," in *Proc. Workshop Historical Document Image Process. (HIP)*, 2015, pp. 123–128.

[16] S. He, P. Samara, J. Burgers, and L. Schomaker, "Image-based historical manuscript dating using contour and stroke fragments," *Pattern Recognit.*, vol. 58, pp. 159–171, Oct. 2016.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1. no. 1, pp. 886–893.

[18] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.

[19] S. He, P. Samara, J. Burgers, and L. Schomaker, "Discovering visual element evolutions for historical document dating," in *Proc. Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, 2016.

[20] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 923–930.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[22] X. Zhang and C. L. Tan, "Handwritten word image matching based on heat kernel signature," *Pattern Recognit.*, vol. 48, no. 11, pp. 3346–3356, Nov. 2014.

[23] M. Rusiñol and J. Lladós "Boosting the handwritten word spotting experience by including the user in the loop," *Pattern Recognit.*, vol. 47, no. 3, pp. 1063–1072, Mar. 2014.

[24] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[25] S. He, M. Wiering, and L. Schomaker, "Junction detection in handwritten documents and its application to writer identification," *Pattern Recognit.*, vol. 48, no. 12, pp. 4036–4048, Dec. 2015.

[26] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, nos. 285–296, pp. 23–27, 1975.

[27] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognit.*, vol. 33, no. 2, pp. 225–236, Feb. 2000.

[28] R. F. Moghaddam and M. Cheriet, "AdOtsu: An adaptive and parameterless generalization of Otsu's method for document image binarization," *Pattern Recognit.*, vol. 45, no. 6, pp. 2419–2431, Jun. 2012.

[29] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Commun. ACM*, vol. 27, no. 3, pp. 236–239, Mar. 1984.

[30] K. Liu, Y. S. Huang, and C. Y. Suen, "Identification of fork points on the skeletons of handwritten Chinese characters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 1095–1100, Oct. 1999.

[31] S. He and L. Schomaker, "A polar stroke descriptor for classification of historical documents," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 6–10.

[32] A. A. Brink, J. Smit, M. L. Bulacu, and L. R. B. Schomaker, "Writer identification using directional ink-trace width measurements," *Pattern Recognit.*, vol. 45, no. 1, pp. 162–171, Jan. 2012.

[33] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[34] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[35] E. Roman-Rangel, C. Pallan, J.-M. Odobez, and D. Gatica-Perez, "Analyzing ancient maya glyph collections with contextual shape descriptors," *Int. J. Comput. Vis.*, vol. 94, no. 1, pp. 101–117, Aug. 2011.

[36] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.

[37] D. Huang, C. Zhu, Y. Wang, and L. Chen, "HSOG: A novel local image descriptor based on histograms of the second-order gradients," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4680–4695, Nov. 2014.

[38] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, Sep. 1990.

[39] L. Schomaker and M. Bulacu, "Automatic writer identification using connected-component contours and edge-based features of uppercase western script," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 787–798, Jun. 2004.

[40] T. Mensink, E. Gavves, and C. G. M. Snoek, "COSTA: Co-occurrence statistics for zero-shot classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 2441–2448.

[41] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2014.

[42] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun. 2005, pp. 524–531.

[43] T. Van der Zant, L. Schomaker, and K. Haak, "Handwritten-word spotting using biologically inspired features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1945–1957, Nov. 2008.

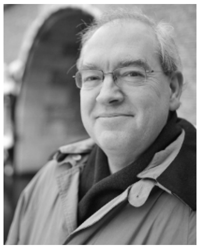
[44] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 701–717, Apr. 2007.

[45] T. Ojala and M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[46] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.

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

Petros Samara received the M.S. degree in philosophy from Erasmus University, Rotterdam, in 2002, and the M.S. degree in medieval history from the Free University of Amsterdam in 2005. He is currently a Visiting Scholar with the Huygens Institute of the History of the Netherlands, Hague, where he is involved in 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 received a degree in medieval history from the University of Amsterdam, and his Ph.D. (*cumlaude*) in 1993 on the Palaeography of the documentary sources of Holland and Zeeland in the thirteenth century. He is currently a Senior Researcher at the Huygens Institute of the History of the Netherlands, The Hague, and a parttime Professor at the University of Amsterdam. He has authored over 120 publications, mainly on palaeography, diplomatics, medieval chronicles, and source editions.

Lambert Schomaker (SM'–) is currently a Full Professor with the Artificial Intelligence at the University of Groningen. He has been the Director of AI Institute ALICE since 2001. His main interest is in pattern recognition and machine learning problems, with applications in handwriting recognition problems. He has authored 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. He is a member of the IAPR and 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.