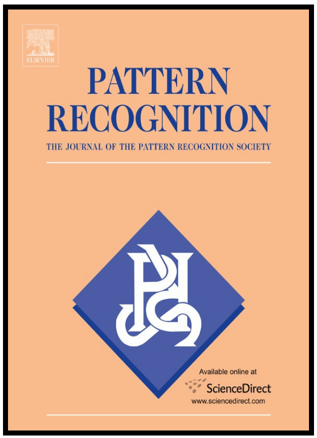


www.elsevier.com/locate/pr

Beyond OCR: Multi-faceted understanding of handwritten document characteristics

Sheng He, Lambert Schomaker

PII: S0031-3203(16)30278-3
DOI: http://dx.doi.org/10.1016/j.patcog.2016.09.017
Reference: PR5879

To appear in: *Pattern Recognition*

Received date: 10 May 2016
Revised date: 19 September 2016
Accepted date: 20 September 2016

Beyond OCR: Multi-faceted understanding of handwritten document characteristics

Sheng He*, Lambert Schomaker

Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen, PO Box 407, 9700 AK, Groningen, The Netherlands

Abstract

Handwritten document understanding is a fundamental research problem in pattern recognition and it relies on the effective features. In this paper, we propose a joint feature distribution (JFD) principle to design novel discriminative features which could be the joint distribution of features on adjacent positions or the joint distribution of different features on the same location. Following the proposed JFD principle, we introduce seventeen features, including twelve textural-based and five grapheme-based features. We evaluate these features for different applications from four different perspectives to understand handwritten documents beyond OCR, by writer identification, script recognition, historical manuscript dating and localization. Extensive experimental results demonstrate that our novel QuadHinge and CoHinge features following the JFD principle provide promising results on these four applications.

Keywords: handwritten document understanding, joint feature distribution principle, writer identification, script identification, historical manuscript dating and localization

*Corresponding author

Email addresses: heshengxgd@gmail.com (Sheng He), L.Schomaker@ai.rug.nl (Lambert Schomaker)

1. Introduction

Nowadays, a large number of historical manuscripts have been digitized and technologies from pattern recognition have been applied to handwritten text search and retrieval. Automatical reading of the text context by OCR methods is not enough to completely understand the handwritten manuscripts. In practice, current OCR systems are actually far from perfect anyway, such that additional image information should be used in order to understand more of a given document. For example, writer, date and geographical prevenance (location) are very important to correctly understand the valuable historical information contained in manuscripts by historians and paleographers [1].

Handwriting can be used as human behavioral biometrics measure [2] as the individual handwriting style is encoded into the handwritten patterns when they were written down. This allows for the analysis of the handwriting style of manuscripts based on handwritten texts to uncover the important context information, such as the writer, date and location. The main task of handwriting style analysis is to design handwriting style-specific features to extract the visual attributes of writing. Feature representation maps the raw pixels of characters into a discriminant high-dimensional space [3, 4] which captures specific information of the characters and can be processed by computers and it takes a very important role in pattern recognition and computer vision field. In fact, many efforts have been made to design discriminative and powerful features in computer vision [4]. Although it is shown that a (deep) learning-based feature representation may achieve significant results in various applications, the hand-crafted features are still very important in handwritten document analysis because the amount of data in historical manuscript collection is usually not big enough to train deep neural networks. For example, the ImageNet data set [5] contains several millions samples for training while most historical books contain only several thousands pages. Particularly, historical manuscripts from ancient times are very rare and it is hard to use a complex data-intensive machine learning methods to learn a shallow model which, with disigned features, may be used on small collections. By harvesting classess using shallow methods, deep learning can be applied after a critical mass of data has been harvested.

In this paper, we propose a general joint feature distribution (JFD) principle, which allows researchers to design more powerful and discriminative features based on the existing feature extraction methods. Several novel features are proposed following the JFD principle, such as the CoLBP inspired by the co-occurrence pattern distributions [6], CoHinge and QuadHinge based on the original Hinge kernel [2], and the Ink Context inspired by the junction feature [7] and shape context [8]. We apply the existing and proposed features for multi-faceted understanding of handwritten manuscripts beyond OCR and evaluate these features from four perspectives: answering 4W questions in paleography and book history [1]: *Who*, *Which*, *When* and *Where*, corresponding to writer identification, script identification, dating and localization problems which can describe the historical context of manuscripts. Fig. 1 shows the five important questions with the OCR problem and their corresponding research problems to understand handwritten manuscripts.

Although there are many methods proposed in the literature for writer and script identification, very little work has been done on the evaluation of the performance of these features on both writer and script identification, and manuscript dating and localization. Therefore, it is still very hard for historians or paleographers to choose the appropriate features to perform specific tasks on their own data sets. Our work in this paper, therefore, is to provide a comprehensive performance evaluation of different features for different applications and to provide new perspectives and insights for feature designing.

The rest of this paper is organized as follows. In Section 2, we introduce the joint feature distribution (JFD) principle. Section 3 presents seventeen features, including twelve textural-based and five grapheme-based features. Section 4 provides the extensive experimental results for writer identification, script identification, historical manuscript dating and localization and the discussion and conclusion is presented in Section 5.

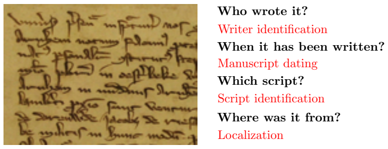


Figure 1: The four interesting questions for handwritten manuscript understanding and their corresponding problems beyond OCR.

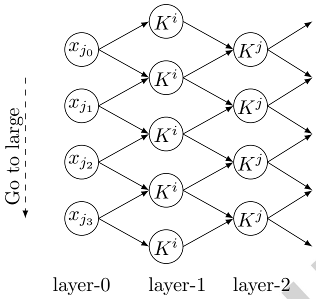


Figure 2: Feature network: more powerful and discriminative features can be generated following the four JFD principles. x_{j_z} is the position on the image and K^i and K^j are the kernel functions (they can be the same or different type of kernel functions). Each note can be described by single feature $f^i(x_{j_z})$ or by the joint feature $f^{i,\cdots,i+n}(x_{j_z})$. For example, notes in layer-1 can be represented by: $f(K^i(x_{j_z})) = K^i\big(f(x_{j_z}), f(x_{j_{z+1}})\big)$, and notes in layer-2 can be represented by: $f(K^j(x_{j_z})) = K^j\big(f(K^i(x_{j_z})), f(K^i(x_{j_{z+1}}))\big)$.

2. Joint feature distribution principle

Previous studies [2, 6, 9, 10, 11, 12] have shown that the use of spatial co-occurrence among features is more discriminative and powerful. In this paper, we extend this idea to the joint feature distribution principle (JFD principle), which can be divided into three different groups: the spatial joint feature distribution (JFD-S), the attribute joint feature distribution (JFD-A) and the joint kernel feature distribution (JFD-K).

We denote by $f^i(x_j)$ the local feature f^i on the position x_j in an image. Following by the JFD-S principle, new features can be derived as:

$$f^i(x_j, \cdots, x_{j+n}) = \big[f^i(x_j), \cdots, f^i(x_{j+n})\big]_{joint} \tag{1}$$

where $x_j, \cdots, x_{j+n}$ are $n+1$ points on the image which have a certain spatial relationship and the new joint feature $f^i(x_j, \cdots, x_{j+n})$ captures more complex local structures with a larger supporting region.

Several feature methods followed the JFD-S principle have been proposed in the literature. For example, the Gray-Level Co-occurrence Matrices (GLCM) has been proposed for texture classification [9] and writer identification [13]. Pairwise local features has been studied for food recognition in [10] and co-occurrence of histogram of orientation gradient (CoHOG) has been studied in [11].

Following by the JFD-A principle, new features can be derived as:

$$f^{(i,\cdots,i+n)}(x_j) = [f^i(x_j), \cdots, f^{i+n}(x_j)]_{joint} \tag{2}$$

where $f^i(x_j)$ and $f^{i+n}(x_j)$ are different local features on the point x_j which may capture different attributes or properties. The attribute joint feature $f^{(i,\cdots,i+n)}(x_j)$ usually has specific meanings. For example, the joint distribution of ink trace and ink width can capture the property of writing instruments [12]. In [14], the oriented Basic Image Feature Columns (oBIF Columns) which is the joint distribution of six Derivative-of-Gaussian filters at two scales has been applied for writer identification.

Following by the JFD-K principle, new features can be derived as:

$$\begin{gathered} f^i(x_j, \cdots, x_{j+n}, K) = \\ \Big[K\big(f^i(x_j), f^i(x_k)\big), \cdots, K\big(f^i(x_{j+n}), f^i(x_{k+n})\big)\Big]_{joint} \end{gathered} \tag{3}$$

where $K(\cdot)$ is the kernel function which can describe the relationship between feature f^i on two different positions x_j and x_k. Any kernel functions can be chosen and the features with different kernel functions has different properties. For example, using the differential kernel operator based on the Hinge feature [2] derives the Δ^nHinge feature [15] which is a rotation-invariant feature for writer identification.

The difference between spatial joint feature $f^i(x_j, \cdots, x_{j+n})$ and attribute joint feature $f^{(i,\cdots,i+n)}(x_j)$ is that the spatial joint feature $f^i(x_j, \cdots, x_{j+n})$ is the joint distribution of the same type of feature f^i on different positions

$x_j, \cdots, x_{j+n}$ and the attribute joint feature $f^{(i,\cdots,i+n)}(x_j)$ is the joint distribution of different types of features $f^i, \cdots, f^{i+n}$ on the same position x_j. The $f^i(x_j, \cdots, x_{j+n})$ inherits the properties of the $f^i(x_j)$ feature. For example, if $f^i(x_j)$ is sensitive to the rotation changes, the $f^i(x_j, \cdots, x_{j+n})$ is also sensitive to the rotations. However, using kernel functions, $f^i(x_j, \cdots, x_{j+n}, K)$ can introduce new properties or solve the transform invariant problems, depending on the definition of the kernel function K.

One problem of the features derived based on the JFD principle is that the feature dimension is very high. For example, if the dimension of $f^i(x_j)$ is m, the dimension of the joint feature $f^i(x_j, \cdots, x_{j+n})$ is m^{n+1}. Therefore, n is usually set to 1, which results in the spatial co-occurrence features [6, 9, 10]. When n is large, the $f^i(x_j, \cdots, x_{j+n})$ describes large and complex structures in handwritten documents, which are called allographs or graphemes. The distribution of the allographs of documents is sparse in the feature space, which can be solved by the bag-of-word model [16] and the textural-based feature becomes the grapheme-based feature.

Following the proposed three principles, a feature network can be built, as shown in Fig. 2. Each node in the feature network represents the location on the image and can be described by the single feature $f^i(x_j)$ or by the joint feature $f^{(i,\cdots,i+n)}(x_j)$. Recursively using these three principle with proper local features and kernel functions, new and more abstract features can be derived directly from this feature network. For example, given the local feature $f^0(x_i)$, a new feature $f^1(x_i)$ can be built using kernel function K^i by: $f^1(x_i) = K^i\big(f^0(x_i), f^0(x_j)\big)$ where x_i and x_j are spatially adjacent. The new feature $f^1(x_i)$ can be also considered as the local feature $f^0(x_i) = f^1(x_i)$ to build more features with the same or a different kernel functoin K^j. The Δ^nHinge feature [15] is a typical example, where the Δ^nHinge kernel can be computed directly from the Δ^{n-1}Hinge kernel with the differential operator kernel function (see the Δ^nHinge feature in Section 3).

3. Feature representation

In this section, we introduce several typical features developed in the literature for handwritten document analysis. In addition, we also propose several new features followed the JFD principles. The features can be roughly categorized into two groups [2]: textural-based and grapheme-based methods and the computation details are presented in the following sections.

3.1. Textural-based features

Textural-based method considers the handwritten document as a textural image and extracts statistical information from text blocks on the entire image. Textural features extracted from handwritten images often capture the curvature and slant attributes of handwriting style and they usually do not need any segmentation method. Several typical textural-based features in the literature and their extensions are described in this section.

Local Binary Pattern (LBP) [17] LBP is a gray-scale invariant textural feature and is widely used in texture recognition [18] and writer identification [19, 20]. For a pixel x_i in an image, the LBP code is defined as:

$$\mathrm{LBP}_{P,R}(x_i) = \sum_{p=0}^{P-1} s(g_p - g_{x_i}) \cdot 2^p \tag{4}$$

$$s(x) = \begin{cases} 1, & if \quad x \geq 0 \\ 0, & if \quad x < 0 \end{cases} \tag{5}$$

where g_{x_i} and g_p are the pixel values of point x_i and its neighbors, and P and R are the number of neighbors and the radius of the neighbor pixels to the x_i, respectively. Following works [17, 20], we set $P = 8$ and $R = 1$ and we use LBP for short to represent $\mathrm{LBP}_{8,1}$ thereafter. Finally, the 255 patterns without the background one are considered to build the LBP histogram and the resulting descriptor is of dimension 255. LBP follows the JFD-S principle, which joints the binary test $s(x)$ on the eight neighbors of the certain pixel x_p: $\mathrm{LBP}(x_p) = [s(x_1 - x_p), ..., s(x_i - x_p), ..., s(x_8 - x_p)]_{joint}$.

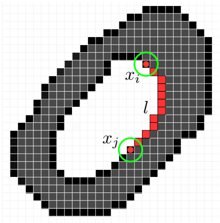


Figure 3: Co-occurrence patterns on ink contours.

Co-occurrence Local Binary Pattern (CoLBP) Following the JFD-S principle, we propose the co-occurrence LBP on handwritten documents, inspired by the work [6]. Given two pixels x_i and x_j with a Manhattan distance l along the ink contour and their LBP uniform codes LBP(x_i) and LBP(x_j) (see Fig. 3), the CoLBP is defined as

$$\text{CoLBP}(x_i, x_j) = \big[\text{LBP}(x_i), \text{LBP}(x_j)\big]_{joint} \tag{6}$$

The uniform LBP code is defined as the binary pattern where there is at most 2 bitwise transitions from 1 to 0 or vice versa [17]. The reason that we consider the uniform LBP code is that the most of LBP codes obtained along the ink contours are uniform patterns. In order to make CoLBP rotation-invariant, we only consider the non-redundant patterns $\big(\text{LBP}(x_i) \leq \text{LBP}(x_j)\big)$. Finally, a 2D histogram is built to represent the probability of the co-occurrence LBP patterns along the ink contours and the dimension of the feature vector is 58*(58+1)/2 = 1711. The parameter l is empirically set to 8 in our experiments.

Run-length Histogram (RLH) Run-length features are widely used in handwritten document analysis [21, 22, 23]. The run-lengths of certain patterns along a given direction, such as '0' and '1' on binarized images, are quantized into a histogram as the feature representation. Usually, run-length histograms of the ink and background pixels with the maximum length 100 are obtained on the horizontal and vertical directions and concatenated together as the final feature vector with the dimension of 2*2*100 = 400. The run-length feature follows the JFD-K principle, where the kernel function is defined to count the length of runs on the scanning direction.

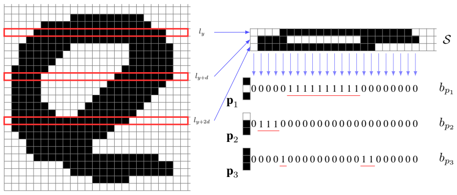


Figure 4: Illustration of run-lengths of local binary pattern $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$ on the sequence $\mathcal{S}$ formed by three parallel scanning lines l_y, l_{y+d} and l_{y+2d} with $d = 6$.

Run-lengths of Local Binary Pattern (LBPruns) The LBPruns feature has been proposed in our previous work [24], which computes the run-lengths of local binary patterns formed with n parallel scanning lines along a given direction with inter-line distance d on binarized images. Fig. 4 provides an example of LBPruns with three lines on the horizontal direction. The number of n determines the number of possible local binary patterns and the inter-line distance d determines the spatial resolution of local binary patterns. Finally, $2^n \times 2$ histograms on horizontal and vertical directions can be obtained and they are concatenated together to form the final feature vector. In this paper, we empirically set n and d to 5 and more detailed information of the selection of these parameters can be found in [24]. The maximum length N_{max} is set to 100 (the discussion of the parameter is presented in Section 4.1.2) and the dimension of the final feature vector is $2*2^5*100 = 6400$. The LBPruns feature follows the JFD-N principle, which builds the feature vector using the run-length methods based on the LPB computations.

Hinge [2] The Hinge feature is the joint probability distribution of the orientations of legs of two contour fragments attached at a common end pixel on the ink contours. Fig. 5 shows two examples of the Hinge kernel on contour fragments with leg length l and the joint probability of the two orientations, α and β ($\alpha < \beta$), are quantized into a 2D histogram. In this paper, we set $l = 7$ and the number of bins of α and β is set to 23. Finally, the dimension of the feature vector is 253. The Hinge feature follows the JFD-A principle, which considers two different directions (can be considered as two attributes) on each contour pixel.

Co-occurrence Hinge (CoHinge) [25] We provide a feature followed the JFD-S principle: the CoHinge, which is defined as the joint distribution of Hinge kernel on two different points x_i and x_j with Manhattan distance l on the contours, similar as the CoLBP feature:

$$\text{CoHinge}(x_i, x_j) = \big[\text{Hinge}(x_i), \text{Hinge}(x_j)\big] \tag{7}$$

Each Hinge kernel has two values α and β, and therefore, the CoHinge kernel has four values $[\alpha(x_i), \beta(x_i), \alpha(x_j), \beta(x_j)]$, which can be quantized into a 4D histogram. The Manhattan distance l is set to 7 (more information of this parameter is shown in Section 4.1.2). We set the number of bins of the angle to 10, and finally the dimension of the CoHinge feature is $10*10*10*10 = 10,000$.

Δ^n*Hinge:* The Δ^nHinge is a rotation-invariant texture features, which has been proposed in [15]. The Δ^nHinge feature can be computed from the feature network (see Fig. 2), with the differential operator between Hinge kernels as the kernel function K:

$$\begin{aligned} f^n(x_i) &= K\big(f^{n-1}(x_i), f^{n-1}(x_i + \delta l)\big) \\ &= K\big(f^{n-1}, \cdot\big) \end{aligned} \tag{8}$$

where $f^n(x_i) = (\Delta^n \alpha, \Delta^n \beta)$ is the Hinge kernel and n is the order of the differential operator. We use $K\big(f^{n-1}, \cdot\big)$ for short representation and the Δ^nHinge can be recursively computed by:

$$\begin{aligned} f^n(x_i) &= K\big(f^{n-1}, \cdot\big) \\ &= K\Big(K\big(f^{n-2}, \cdot\big), \cdot\Big) \\ &= K\bigg(K\Big(K\big(f^{n-3}, \cdot\big), \cdot\Big), \cdot\bigg) \\ &= \cdots \end{aligned} \tag{9}$$

where $f^0 = (\alpha, \beta)$ is the original Hinge kernel [2]. More precisely, the Δ^nHinge kernel is defined as:

$$\begin{cases} \Delta^n \alpha(x_i) = & \frac{\Delta^{n-1}\alpha(x_i) - \Delta^{n-1}\alpha(x_i + \delta l)}{\delta l} \\ \Delta^n \beta(x_i) = & \frac{\Delta^{n-1}\beta(x_i) - \Delta^{n-1}\beta(x_i + \delta l)}{\delta l} \end{cases} \tag{10}$$

Although many different features can be generated based on the feature network with different n, in this paper, we only report the performance of the Δ^1Hinge feature and the feature dimension is 780.

Triple Chain Code (TCC) [26] The chain code on a pixel of the writing contours is the one of eight directions where the next pixel on, denoted from 1 to 8. Following the JFD-S principle, we evaluate the performance of triple chain code (TCC) feature which is also used in [26] for writer identification.

$$\mathrm{TCC}(x_i, x_{i+l}, x_{i+2l}) = [\mathrm{CC}(x_i), \mathrm{CC}(x_{i+l}), \mathrm{CC}(x_{i+2l})] \tag{11}$$

where $\mathrm{CC}(x_i) \in \{1, 2, \cdots, 8\}$ is the chain code value on position x_i, and l is the Manhattan distance along the writing contours. In this paper, we set l to 7, the same as the value of the CoHinge feature. Finally, the feature dimension is $8 \times 8 \times 8 = 512$.

Quill and QuillHinge [12] The Quill feature is the joint probability distribution $p(\alpha, w)$ of the relation between ink direction α and the ink width w, which captures the writing instrument property. It follows the JFD-A principle, because the feature types, the ink direction and ink width, are different. The QuillHinge is an extension of the Quill and Hinge, and it is the probability of $p(\alpha, \beta, w)$, resulting in a 3D histogram. We use the same parameters of the Quill and QuillHinge as the original paper [12], and the dimensions of Quill and QuillHinge are 1600 and 31,200, respectively.

Quadruple Hinge (QuadHinge) [25] We also provide the QuadHinge feature to demonstrate the powerful of the features following the JFD-A principle. In order to incorporate the curvature information of the contour fragments in the Hinge kernel, we define a fragment curvature measurement (FCM) $\mathcal{C}(\mathcal{F}_c)$ for contour fragments, inspired by [27]:

Definition. Let $\mathcal{F}_c$ be a contour fragment on the ink trace, $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$ are the Cartesian coordinates of the two end points. Then the fragment curvature measurement $\mathcal{C}(\mathcal{F}_c)$ is defined as the proportion of the Euclidean distance $\mathbf{d_2}(p_1, p_2)$ between two end points to the length of the contour

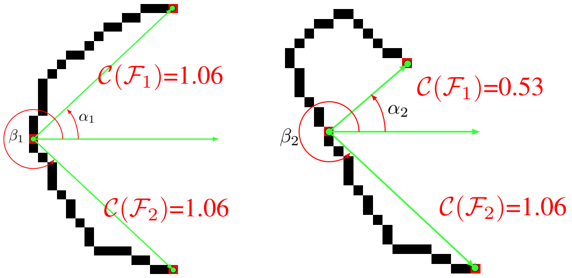

$\mathcal{C}(\mathcal{F}_1)$=1.06
α_1
β_1
$\mathcal{C}(\mathcal{F}_2)$=1.06
$\mathcal{C}(\mathcal{F}_1)$=0.53
α_2
β_2
$\mathcal{C}(\mathcal{F}_2)$=1.06

Figure 5: The left and right figures show two contour fragments with the same Hinge kernel (α_1=α_2 and β_1=β_2) but different fragment curvature values $\mathcal{C}(\mathcal{F}_c)$.

fragments **s**.

$$\mathcal{C}(\mathcal{F}_c) = \mathbf{d_2}(p_1, p_2)/\mathbf{s} \tag{12}$$

where $\mathbf{d_2}(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. Note that all the pixels of the contour fragments are represented in the Cartesian coordinates and the Chebyshev distance between two neighbor pixels is equal to 1.

A novel Quadruple Hinge kernel, which integrates the $\mathcal{C}(\mathcal{F}_c)$ into the original Hinge kernel, defined as: $\mathcal{H}(p, \mathbf{s}) = \{\alpha,\ \beta,\ \mathcal{C}(\mathcal{F}_1),\ \mathcal{C}(\mathcal{F}_2)\}$, where p is the center point, $\mathbf{s}$ is the fragment length, $\mathcal{C}(\mathcal{F}_1)$ and $\mathcal{C}(\mathcal{F}_2)$ are the fragment curvature measurements of the two contour fragments, respectively (see the examples in Fig. 5). Adding the curvature information of the two contour fragments can improve the discriminative of the Hinge kernel. For example, the Hinge kernels $\{\alpha,\ \beta\}$ of the left and right fragments in Fig. 5 are the same. However, the curvatures of the fragments are different, yielding different Quadruple Hinge kernels. Finally, the Quadruple Hinge kernels on all contour pixels are collected and quantized into a 4-D histogram. In order to capture the scale information, we agglomerate the Quadruple Hinge kernels with multiple scales, where the scale factor $\mathbf{s}$ is set as: $\mathbf{s} = \mathbf{s}_0 * (t + 1)$. The $\mathbf{s}_0$ is the basic fragment length and $t = \{0, 1, \cdots, T\}$ and T is the index of maximal scales. In this paper, we set the number of angle bins N_a to 12, the number of $\mathcal{C}(\mathcal{F})$ bins N_l to 6 $\mathbf{s}_0 = 5$ and $T = 10$, and the final feature dimension is $6 * 6 * 12 * 12 = 5,184$. More information of the selection of these parameters can be found in Section 4.1.2.

Cloud Of Line Distribution (COLD) Inspired by the fact that writing contours can be approximated by a set of line segments obtained by the sequential polygonization algorithm [26] and the lengths and orientations of these straight lines can capture the handwriting styles, we design a new curvature-free features called Cloud of Line Distribution (COLD). The COLD feature is the joint probability distribution of the length and orientation of these lines followed the JFD-A principle and the computational procedure is described as follows (see Fig. 6): First, the ordered high-curvature points on the writing contours are obtain using the method [28], denoted by $\mathcal{P} = \{p_i(x_i, y_i), i = 0, 1, 2, \cdots, n\}$, where (x_i, y_i) is the coordinate of the point p_i. The line segments can be obtained between any pair of the dominant points (p_i, p_{i+k}), where k the parameter which denotes the distance on the dominant sequence $\mathcal{P}$. Each line can be measured by a pair (θ, ρ) in the polar coordinate space, where θ is the line orientation and ρ is the line length. All the lines in a given handwritten document can form a distribution in the polar coordinate space and can be quantized into a log-polar histogram inspired by the Shape Context [8]. The features obtained with $k = 1, 2, 3$ in the log-polar space with the radius 7 and the angular intervals 12 are concatenated into one feature vector with the dimension: $7 * 12 * 3 = 252$.

3.2. Grapheme-based features

Grapheme-based features capture the statistical distribution of the allograph segmented from the handwritten texts and it is assume that individuals have their own prototypes in their brain to draw characters. Although any spatial co-occurrence features can be generalized to the grapheme-based features, in this section, we introduce several typical grapheme-based features for handwritten manuscript understanding. In fact, all the grapheme-based features follow the JFD-S principle, which concatenate the spatial information together to obtain a large structure of the ink trace.

Connected-Component Contours (CO^3) [29] The CO^3 is the contour obtained from each connected component of the binarized handwritten images. In order to measure the similarity between CO^{3}s, each CO^3 is resampled to contain

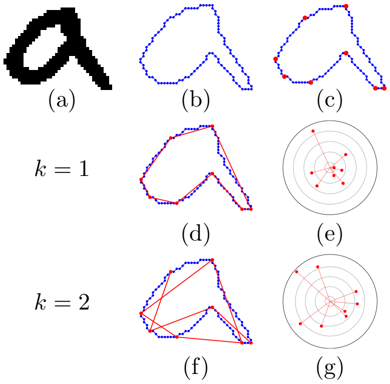


Figure 6: Illustration of the process of the COLD construction: (a) The given binarized connected component; (b) The contour extracted from the binarized image (a); (c) Detected dominant points (red points); (d) Line segments (red lines) obtained between pair dominant points when $k = 1$; (e) The distribution of lines from (d) in the polar coordinate space; (f) Line segments when $k = 2$ (Note that some long lines are not shown in order to make the figure more clear); (g) The distribution of lines from (f) in the polar coordinate space.

a fixed number of coordinate pairs (x_i, y_i) and the normalized coordinate pairs can be considered as the feature vector.

$$\begin{cases} \overline{x_i} = & (x_i - \mu_x)/\sigma_x \\ \overline{y_i} = & (y_i - \mu_y)/\sigma_y \end{cases} \tag{13}$$

where the μ_x and μ_y are averages of the x_i and y_i coordinates and the σ_x and σ_y are the corresponding standard deviations. In this paper, we sample 100 points on each contour and size of the contour descriptor is 2*100 = 200.

k Contour Fragments (kCF) [30] One limitation of the CO^3 is that it is sensitive to the cursive handwriting where characters are always touched with each other and the resulting CO^{3}s are very large and less repeatable. In order to solve such problem, we extract the contour fragments from the contours, inspired by [31]. The dominant points $\mathcal{P} = \{p_i(x_i, y_i), i = 0, 1, 2, \cdots, n\}$ are obtained first, using the same method as in the COLD feature. We compute the break points $\mathcal{B} = \{b_i(x_i, y_i), i = 0, 1, 2, \cdots, n\}$ as the midpoints of each pair of dominant point (p_i, p_{i+1}), and the contour fragments can be obtained between any pair of the break points (b_i, b_{i+k}), denoted as kCF. Similar as the CO^3

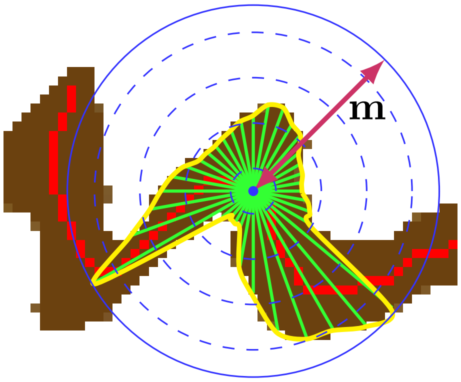
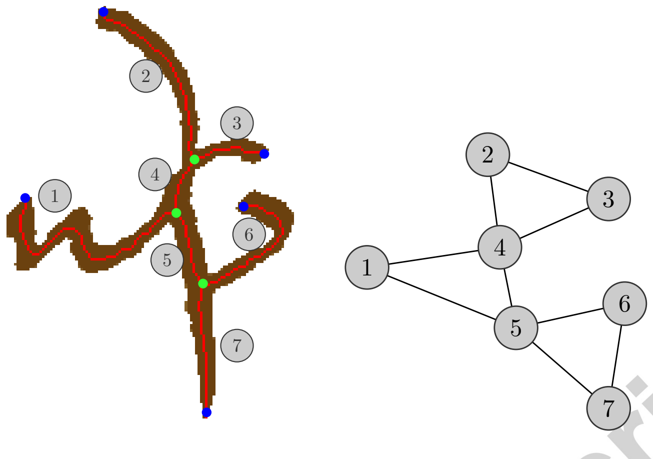


Figure 7: An illustration of the stroke-length distribution on a reference point (the blue point in the center). The green rays are the partial length in each direction, and the yellow curve is the distribution of the partial length in the polar space. The red line is the skeleton line of the stroke ink. **m** is the maximum measurable stroke length.

method, 100 points are sampled from each contour fragment and normalized using the method described in Eq.(13) to describe each kCF.

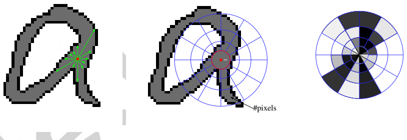
Junction features (Junclets) [7] Junction feature is the stroke-length distribution in every directions from 0 to 2π around a reference point (see Fig. 7) inside the ink trace. When the center point lies on the junction points, such as the fork points and high curvature points on the skeleton line of the ink strokes, the corresponding feature is the junction feature, which contain the junction information around the joint point. In this paper, we compute the stroke length distribution in 120 directions equidistantly sampled from 0 to 2π and the feature dimension of each junction is 120.

k Stroke Fragments (kSF) [30] The connected component of handwritten texts can be decomposed into fragments based on the fork points. Fig. 8 shows an example of a connected component and seven primary stroke fragments are obtained by segmenting at the fork points, which are denoted by numbers 1 to 7. In order to extract longer and more complex stroke fragments, a stroke fragment graph (SFG) is built where the nodes correspond to the primary strokes and two nodes are linked if their corresponding stroke fragments connect to each other (An example is shown in Fig. 8(b)). From the SFG, we can obtain the



(a) (b)

Figure 8: Fig.(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. Fig.(b) shows the corresponding stroke fragment graph (SFG).



(a) (b) (c)

Figure 9: Figure (a) shows the stroke length distribution in the directions θ_m from 0 to 2π. The red point is the fork point and blue lines are the stroke length. Figure (b) shows the scale-invariant log-polar space on the fork point. The scale factor w is equal to the half of the stroke width on the fork point. The ink context is built to count the number of ink pixels in each bin. Figure (c) shows the resulting descriptors.

more complex stroke fragments (kSF) from any connected sub-graph in the SFG with the path length k without any loops. Then N_s reference points are sampled equidistantly on the skeleton line of the stroke fragment and each point is described by the junction features. Finally, all the N_s junction features concatenated together to form the final feature vector. In this paper, we set N_s to 10 and the size of kSF descriptor is 120*10=1,200.

Ink Context (IC) Given a reference point inside the ink, a scale-invariant log-polar space is built with the scale factor w, which is the half stroke width on the reference point (see Fig. 9). The scale factor w can be computed by $w = min\{len(\theta_m)\}$, where $len(\theta_m)$ is the stroke length in the direction θ_m. Inspired by the Shape Context [8], a coarse histogram is computed by counting the number of ink pixels in each bin of the log-polar space. In this paper, we set the parameters of the log-polar space as: the radius is set 4 and the angular intervals is set 120. Finally, the size of the IC histogram is 4*120 = 480.

For all the five grapheme-based features, we randomly select handwritten documents to train the codebook using the 2D Kohonen Self-Organizing Map (SOM) [2, 32] with the cell size $30 \times 30 = 900$.

4. Applications

In this section, we evaluate the twelve textural-based features and five grapheme-based features to answer the 4W questions to understand the handwritten manuscript, corresponding to the writer identification, script identification, manuscript dating and localization problems.

4.1. Writer identification

Writer identification is to answer the question: "who wrote the given document?" according to the characteristic handwriting style encoded in the handwritten text and it has been widely studied in the literature [21, 29, 2, 26, 12, 7]. Given the query handwritten document $q_{w_x}^{s_i}$, where s_i is the script of the handwritten text and w_x is the writer which needs to be identified, all the documents in the database $p_{w_i}^{s_i} \in \mathcal{D}^{s_i}$ with labels of writer w_i and script s_j are sorted according to the feature distance between $q_{w_x}^{s_i}$ and $p_{w_i}^{s_i}$ to output a hitlist where the writer of the top document is assigned to w_x. In this paper, three different experimental settings are considered as following:

- Writer identification based on single-script. Both the query document $q_{w_x}^{s_i}$ and documents on the database $p_{w_i}^{s_i} \in \mathcal{D}^{s_i}$ are written with the same script s_i.

- Writer identification based on mixed-scripts. Both the query document $q_{w_x}^{(s_i,s_j)}$ and documents on the database $p_{w_i}^{(s_i,s_j)} \in \mathcal{D}^{(s_i,s_j)}$ are written with two different scripts (s_i, s_j).
- Writer retrieval. For the query handwritten document q^{s_i}, there are more than one documents from the same hand on the database $\mathcal{D}^{s_i}$. For the task of writer retrieval, our aim is to retrieve the list of handwritten documents which are from the same hand with the query document q^{s_i}.

Writer identification is performed in a "leave-one-out" manner [2, 7, 12, 26]: taking the query document out and sorting the rest 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. In this paper, Top-1 and Top-10 are adopted in all experiments. χ^2 distance is used because it is the best distance function for the probability feature vector [2].

4.1.1. Data sets

Several public databases are available for writer identification, such as the Firemaker [33], IAM [34], CERUG [7], ICFHR2012 Arabic data set [35] and ICDAR2013 [36]. The Firemaker set contains four pages of handwriting written by 250 Dutch subjects: page 1 and page 4 contain the lower-case letters, page 2 was written by only uppercase letters, and Page 3 contains the "forged" text. We use the page 1 vs 4 in our experiments, similar as works in [2, 12, 7]. The IAM set contains 650 writers written in English, modified following the work [2] from the original IAM database [34]. The CERUG set is a cross-script data set, written by 105 Chinese subjects on four pages: page 1 and page 2 were written in Chinese, page 3 contains the English text and page 4 contains both Chinese and English characters. The data set used for the ICFHR2012 competition on writer identification with Arabic scripts [35] contains 204 writers and we only use the first two paragraphs to perform writer identification. The data set used for the ICDAR2013 competition on writer identification [36] contains 250 writers

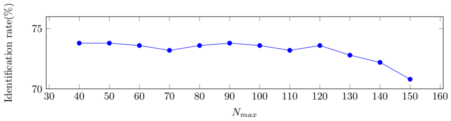


Figure 10: Writer identification performance of the LBPruns feature with different values of the maximum run-length N_{max} on the Firemaker data set.

with four pages (2 English and 2 Greek).

4.1.2. Parameter evaluation

In this section, we present the evaluation of the parameter selection for different features.

Fig. 10 shows the Top-1 performance of writer identification on the Firemaker data set with different maximum run-length N_{max}. From the figure we can find that the performance is quite stable when $N_{max} \in [40, 120]$. As mentioned above, we set it to 100, following the work [22].

Fig. 11 shows the Top-1 performance of writer identification on the Firemaker data set with different Manhattan distance l between two different points x_i and x_j of the CoHinge feature. From the figure we can find that $l = 7$ provides the optimal result.

There are four parameters of the QuadHinge feature: the number of angle bins N_a, the number of curvature $\mathcal{C}(\mathcal{F})$ bins N_l, the maximal scale T and the basic fragment length $\mathbf{s}_0$. Fig. 12 shows the performance of writer identification on the Firemaker data set with different values of the parameter N_a, N_l and T. We can see that the best values are: $N_a = 12$, $N_l = 6$ and $T = 10$. The basic fragment length $\mathbf{s}_0$ should be approximately equal to the average stroke width of the input document. We have found the average stroke width of the whole Firemaker data set is approximately equal to 5. Therefore, we empirically set $\mathbf{s}_0 = 5$.

4.1.3. Performance of writer identification based on single-script

In this section, we evaluate the feature performance for writer identification based on single-script and the results on five data sets are given in Table 1,

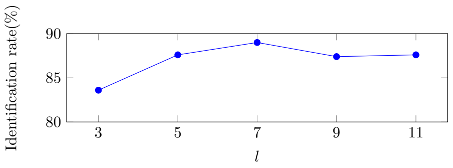


Figure 11: Writer identification performance of the CoHinge feature with different values of the Manhattan distance l between two points on the Firemaker data set.

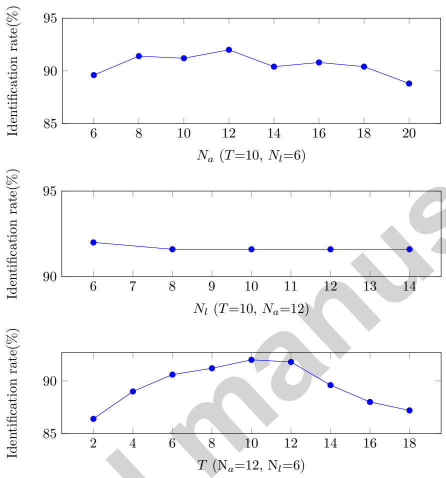


Figure 12: Writer identification performance of the QuadHinge feature with different parameters (the number of curvature bins N_a, the number of angle bins N_l and the value of maximal scales T) on the Firemaker data set.

from which we can see that the textural-based features provide better results than the grapheme-based features. The results of CoLBP are better than LBP. LBPruns provides better results on the five data sets than LBP and RLH, except ICFHR2012 with Arabic handwriting. CoHinge and QuadHinge give the better results than Hinge on all the five data sets. These results demonstrate that the joint feature distribution followed the JFD principle can improve the performance of writer identification.

We can also find that none of these features achieves the best results on all the five data sets. The QuadHinge feature achieves the best results on Firemaker, IAM and CERUG with Chinese data sets, because the QuadHinge cap-

Table 1: The writer identification performances based on single-script on five data sets: Firemaker, IAM, CERUG, ICFHR2012 and ICDAR2013. The column "Dim" is the dimensionality of each feature. The Top-1 identification rates which are greater than 90% are highlighted with gray color.

	Feature	Dim	Firemaker		IAM		CERUG				ICFHR2012		ICDAR2013			
			250 writers		650 writers		105 writers				204 writers		250 writers			
			Dutch		English		Chinese		English		Arabic		English		Greek	
			Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10
Textural-based	LBP	255	51.2	80.2	62.8	83.5	44.8	68.1	11.9	26.7	38.2	77.2	46.8	74.2	53.2	79.8
	CoLBP	1,711	68.8	92.6	66.5	88.0	73.8	95.7	23.3	52.4	56.9	89.9	54.0	85.8	72.2	93.0
	RLH	400	59.6	86.2	71.1	89.0	77.1	92.9	25.7	64.8	51.2	80.1	63.0	90.2	74.6	91.8
	LBPruns	6,400	72.2	91.8	81.4	94.4	88.6	95.7	77.1	98.1	42.9	73.0	83.2	97.6	82.4	97.0
	Hinge	253	84.8	95.8	85.8	95.1	90.9	95.7	22.8	47.1	75.5	92.4	86.2	95.4	80.4	96.2
	CoHinge	10,000	91.6	96.4	92.4	96.5	95.2	98.1	42.8	78.1	**93.6**	**99.3**	93.0	96.2	93.8	**98.8**
	Δ^1Hinge	780	75.6	94.0	82.1	95.7	80.5	93.3	90.5	98.6	61.3	89.5	75.6	93.8	78.6	92.8
	TCC	512	86.2	95.2	88.1	95.6	92.9	97.1	23.8	47.6	85.8	97.8	86.6	94.0	87.6	96.4
	QuadHinge	5,184	**92.2**	**97.2**	**93.2**	96.5	**96.2**	**98.6**	46.7	83.3	87.0	98.3	94.2	96.8	95.2	98.6
	Quill	1,600	69.2	86.8	89.1	95.8	87.1	92.4	28.1	67.6	90.4	**99.3**	92.8	97.2	95.2	97.2
	QuillHinge	31,200	77.4	93.8	89.1	**97.0**	89.0	93.3	**97.1**	**99.0**	85.0	97.1	**95.2**	**98.4**	**96.0**	98.4
	COLD	252	83.0	94.6	83.6	95.9	88.5	97.6	92.4	97.1	61.3	90.4	81.6	93.6	82.0	96.6
Grapheme-based	CO^3	900	56.0	71.8	73.5	88.8	79.0	94.8	75.7	94.8	61.5	86.5	89.8	95.8	90.8	97.8
	kCF	900	67.0	89.0	78.5	91.3	89.0	98.1	77.1	93.8	46.8	79.2	88.2	93.8	86.6	95.6
	Junclets	900	80.2	93.4	85.8	95.5	93.3	98.1	92.9	97.1	56.4	85.5	91.0	96.2	90.4	97.2
	kSF	900	71.8	88.0	74.0	89.0	89.5	95.7	80.0	95.7	33.1	63.7	78.8	94.2	74.8	92.6
	IC	900	77.0	93.8	84.9	95.6	89.0	95.2	91.4	98.1	30.8	63.7	90.4	97.6	92.4	98.2

tures the curvature information of handwriting based on the writing angle and the curvature measure of the contour fragments with a multiple scale strategy. The QuillHinge feature provides the best results on CERUG with English and ICDAR2013 data sets because the handwritten documents on these two data sets were written with different pens and the QuillHinge feature can capture the writing instrument property. However, the dimension of the QuillHinge is also high, which needs more computational time than other features. The second best performance is achieved by COLD on the CERUG data set with English handwriting because the English handwriting written by Chinese subjects contain less curvature [7] and the curvature-less COLD can capture this property. The best result on the ICFHR2012 data set is achieved by the CoHinge feature, and its performance is significantly better than other features.

For the grapheme-based features, the Junclets provides the best results on the Firemaker, IAM, CERUG with Chinese and ICDAR2013 with English data sets. However, CO^3 provides the best Top-1 results on the ICFHR2012 with Arabic data set and the IC feature gives the best results on the Greek handwriting of ICDAR2013 data set.

4.1.4. Performance of writer identification based on mixed-scripts

Table 2: The performance of writer identification based on mixed-scripts on the MIXED data sets. The up-right arrow means the performance increases and down-right arrow means the performance decreases compared to the performance on the single script in Table 1.

Feature	CERUG-MIXED 210 writers Chinese-English		CERUG-Synthetic 210 writers Chinese-English		ICDAR2013-Synthetic 250 writers Greek-English	
	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10
LBP	70.9 ↗	91.9 ↗	33.3 ↘	66.7 ↘	65.4 ↗	87.8 ↗
CoLBP	82.4 ↗	97.6 ↗	61.4 ↘	87.1 ↘	80.8 ↗	96.2 ↗
RLH	47.1 ↘	82.9 ↘	83.3 ↗	95.2 ↗	87.2 ↗	98.0 ↗
LBPruns	91.0 ↗	**100** ↗	92.8 ↗	96.2 ↗	95.6 ↗	99.8 ↗
Hinge	85.2 ↘	95.7 ↘	84.3 ↘	97.6 ↘	92.2 ↗	97.8 ↗
CoHinge	**96.7** ↗	98.6 ↗	89.5 ↘	98.6 ↘	97.6 ↗	98.4 ↗
Δ^1Hinge	88.6 ↘	99.5 ↗	88.1 ↘	97.1 ↘	86.8 ↗	96.2 ↗
TCC	92.9	96.7 ↘	84.2 ↘	95.7 ↘	93.4 ↗	97.8 ↗
QuadHinge	93.8 ↘	97.6 ↘	91.4 ↘	97.6 ↘	97.4 ↗	99.4 ↗
Quill	75.2 ↘	90.9 ↘	73.3 ↘	91.4 ↘	95.8 ↗	99.4 ↗
QuillHinge	85.2 ↘	98.1 ↘	94.8 ↘	98.1 ↘	**99.0** ↗	**100** ↗
COLD	93.8 ↗	**100** ↗	93.3 ↗	97.7 ↗	90.2 ↗	97.2 ↗
CO^3	56.2 ↘	89.5 ↘	89.0 ↗	96.7 ↗	97.8 ↗	99.6 ↗
*k*CF	67.1 ↘	91.0 ↘	94.3	98.1	93.8 ↗	97.8 ↗
Junclets	91.4 ↘	**100** ↗	**95.7** ↗	99.0 ↗	96.0 ↗	99.2 ↗
*k*SF	84.8 ↘	97.1 ↗	94.7 ↗	99.0 ↗	94.0 ↗	98.8 ↗
IC	84.8 ↘	98.6 ↗	94.8 ↗	**99.5** ↗	97.8 ↗	**100** ↗

In this section, we evaluate the performance of writer identification based on the mixed-script handwriting. To our best knowledge, the page 4 of the CERUG data set is the only one real mixed-script data set, which is split into two parts for writer identification, named CERUG-MIXED data set. We also create the synthetic mixed-scripts data set by merging two handwritten documents with different scripts from the same hand into one document. By this way, two synthetic mixed-scripts data sets can be generated: the CERUG-Synthetic data set which contains the synthetic handwritten documents with Chinese and English, and the ICDAR2013-Synthetic data set which contains the synthetic handwritten documents with English and Greek.

Table 2 shows the performance of different features on the three mixed-scripts data sets. The Top-1 performance of the LBP, CoLBP, LBPruns, Co-Hinge and COLD increases on the CERUG-MIXED data set and the performance of others features decreases compared to their performance on the single-script data set shown in Table 1. On the CERUG-Synthetic data set, only the RLH, LBPruns and COLD features provide better results among the textural-based features. The performance of all the grapheme-based features are improved on the the CERUG-Synthetic data set and the performances of all of seventeen features are improved on the ICDAR2013-Synthetic data set. The main reason is that each document on the CERUG-Synthetic and ICDAR2013-Synthetic data sets contains more handwritten texts, which makes the codebook-based features more stable. The LBPruns, COLD and Junclets features reach 100% Top-10 recognition rates on CERUG-MIXED data set and the IC and QuillHinge features reach 100% Top-10 rate on ICDAR2015-Synthetic data set.

Another interesting observation can be found that only the LBPruns and COLD features improve the performance on the three mixed data sets. The reason might be that the LBPruns and COLD features are the curvature-free features which can handle the difference between different scripts.

4.1.5. Performance of writer identification on a large mixed data set

In this section, we give the performance of the seventeen features on a large and mixed data set. Following the work [2], we merge the Firemaker, IAM, CERUG, CVL, ICFHR2012 and ICDAR2013 data sets to obtain a large and combined data set which contains 5177 handwritten documents from 1760 hands with six languages: Dutch, English, Chinese, German, Arabic and Greek. We call this data set as “Large Multi-script Handwritten Set” in the following sections.

Table 3 shows the performances of the different features on this large set.

Table 3: The performance of writer identification on the Large Multi-script Handwritten Set.

Features	Top-1	Top-10	Features	Top-1	Top-10
CoHinge	**93.8**	**97.6**	Δ^1Hinge	78.7	92.5
QuadHinge	92.2	96.9	LBPruns	78.6	92.6
QuillHinge	88.6	96.5	*k*CF	78.1	91.4
TCC	86.4	93.7	CO^3	75.2	88.5
Quill	85.9	93.9	*k*SF	73.5	87.1
Junclets	84.7	94.0	RLH	67.6	87.8
Hinge	82.7	92.6	CoLBP	62.8	81.2
COLD	81.8	94.1	LBP	61.4	81.0
IC	81.2	92.2			

The CoHinge feature achieves the best results and Top-1 recognition rate is 93.8%. The QuadHinge feature also provides a comparable result with 92.2%. which is better than the QuillHinge feature. This indicates that the handwriting styles captured by CoHinge and QuadHinge take more important information than the property of writing instruments captured by the Quill and QuillHinge feature on this large data set where handwritten documents are written with different pens and scripts. For the grapheme-based features, the performances of the Junclets and IC are comparable, which outperform other three methods.

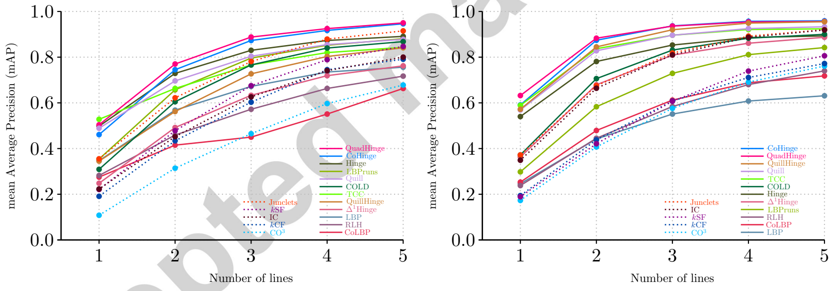
The results of the feature combinations between textural-based and grapheme-based features are presented in Table 4. Generally, combining two features provide an improvement for writer identification [2, 26]. However, from Table 4 we can see that combining CoHinge and QuadHinge with other grapheme-based features provides a worse performance, which demonstrates that the CoHinge and QuadHinge contain the discriminative information of handwriting style and linearly combining them with other grapheme-based features can not introduce more useful information.

4.1.6. Performance of writer retrieval

In this section, we perform the task of writer retrieval based on handwritten documents with different text lines, similar as our work [25]. We segment the handwritten documents into text lines and conduct experiments on handwritten documents with different number of text lines from one to five for writer retrieval. We use the CVL data set [37], which contains handwritten documents

Table 4: The Top-1 performance of writer identification based on dual feature combination on the Large Multi-script Handwritten Set. The recognition rates in bold increase while with italic type decrease, compared to the best performance of the individual features (shown in Table 3) involved in the combination.

	Feature2 / Feature1	Grapheme-based features									
		CO^3		*k*CF		Junclets		*k*SF		IC	
		Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10	Top-1	Top-10
Textural-based features	LBP	**81.9**	**92.7**	**84.0**	**94.5**	**88.2**	**95.9**	**79.2**	**91.8**	**85.3**	**94.8**
	CoLBP	**78.5**	**91.0**	**81.8**	**93.7**	**86.0**	**94.8**	**75.9**	**89.3**	**82.9**	**93.8**
	RLH	**88.3**	**96.3**	**87.7**	**96.4**	**89.3**	**96.9**	**85.2**	**95.3**	**87.7**	**96.8**
	LBPruns	**85.4**	**95.3**	**84.2**	**95.0**	*83.9*	**95.0**	**82.2**	**94.0**	**83.6**	**94.9**
	Hinge	**83.4**	**93.8**	**86.7**	**95.6**	**88.9**	**95.9**	*82.6*	**93.2**	**86.4**	**95.4**
	CoHinge	*90.4*	*96.5*	*93.1*	*97.3*	*92.9*	*97.4*	*91.1*	*97.0*	*91.6*	*97.2*
	Δ^1Hinge	**82.9**	**93.3**	**86.3**	**96.7**	**89.9**	**96.3**	**83.4**	**93.4**	**86.1**	**95.0**
	TCC	*84.0*	**94.3**	**87.7**	**95.7**	**89.5**	**96.3**	*83.8*	*93.6*	**87.5**	**95.6**
	QuadHinge	*87.6*	*95.8*	*91.0*	*96.6*	**92.3**	**97.1**	*89.2*	*96.1*	*90.3*	*96.6*
	Quill	**86.4**	**95.1**	**89.8**	**96.5**	**90.8**	**96.9**	**86.3**	**95.7**	**89.0**	**96.6**
	QuillHinge	**88.7**	*96.0*	**91.3**	**97.1**	**91.4**	**97.1**	**88.9**	*96.4*	**89.7**	**96.7**
	COLD	**87.9**	**96.0**	**89.1**	**96.3**	**90.1**	**96.6**	**86.2**	**95.2**	**89.1**	**96.3**



(a)

(b)

Figure 13: Performance of writer retrieval with respect to amount of text in a sample: (a) writer retrieval on the CERUG-CN data set with Chinese handwriting, and (b) writer retrieval on the CVL data set with English handwriting. Solid lines represent textural-based features and dashed lines represent grapheme-based features. Note that the legend is sorted in descending order of performance.

from 310 writers, 27 of which wrote 7 texts and 283 writers have 5 texts. In this experiment, only English handwriting from 310 writers are considered. In addition, we also evaluate the performance of writer retrieval on the Chinese handwriting from the CERUG data set. Table 5 shows the number of query

Table 5: The number of query samples with different line texts on the CERUG and CVL data sets for writer retrieval.

Data set	line 1	line 2	line 3	line 4	line 5
CERUG 201 writers Chinese	2095	1012	617	450	345
CVL 310 writers English	10820	5038	3187	2118	1686

samples on the two data sets with different text lines.

We use the mean Average Precision (mAP) to measure the performance of writer retrieval, which is defined as:

$$\text{mAP} = \frac{1}{N}\sum_{q=1}^{N} AveP(q) \tag{14}$$

where N is the number query samples and $AveP(q)$ is the average precision of the query q.

Fig. 13 shows the results on the two CERUG Chinese and CVL English data sets. The performance is improved when the number of lines increases. All of the features give the reasonable results on the handwritten documents with at least three lines. The results of the CoHinge and QuadHinge features provide the best performance than other features on the two data sets and the performance of the Junclets feature is higher than other four grapheme-based features.

4.2. Script identification

Script identification is the problem to automatically recognize the script of a given document [38], and it has been widely studied on printed documents [39, 40, 41, 42] and on handwritten documents [43, 44]. Identifying script on handwritten documents is more difficult because the handwritten texts contain not only the script shape information, but also the handwriting styles from

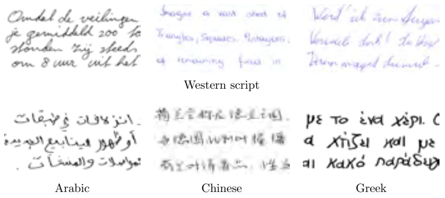


Figure 14: Example of handwriting from four major different languages.

different writers. In this section, we evaluate the seventeen features for script identification. Furthermore, we investigate the relationship of feature performance between writer identification and script identification based on the Large Multi-script Handwritten Set, which is mentioned on the previous section. There are four major different scripts: Chinese, Arabic, Greek and Western scripts (including English, German and Dutch) and Fig. 14 gives an example handwriting of the four different scripts.

Fig. 15 shows the script identification performance with respect to the number of K using the K nearest neighbor (KNN) method. From the figure we can see that the results of all the features decrease slightly when K increases from 5 to 50. The best performance is achieved by the kCF and QuadHinge feature when K=10, which demonstrates that the character shapes take an important role for script identification (a similar observation has also been shown in [44]).

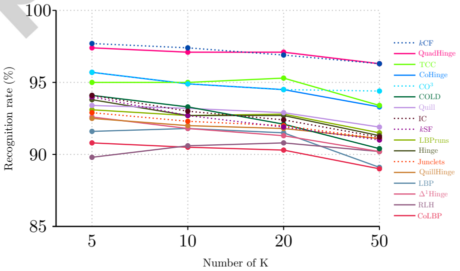


Figure 15: Script identification rates of the seventeen features with respect ot the number of K using the KNN method. Note that the features are sorted with descending order of the performance when K=10.

We also study the feature performance for both writer and script identifica-

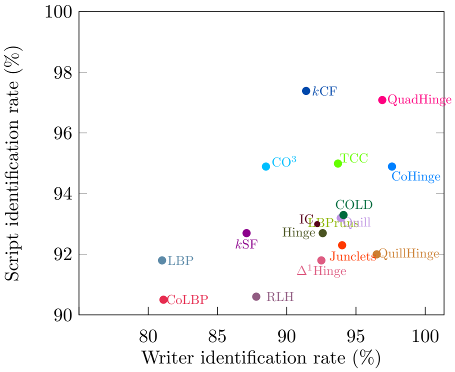


Figure 16: The performance of script recognition with K=10 with respect to the Top-10 performance of writer identification. Note that the features closed to the top-right corner are discriminative both for script identification and writer identification.

tion. Fig. 16 shows the relationship of feature performance of script identification with K=10 and the Top-10 performance of writer identification. From the figure we can see that QuadHinge is more closed to the top-right corner, which means that the QuadHinge feature is discriminative both for script identification and writer identification. In addition, the Δ^1Hinge, Junclets, Quill and COLD features close to the dignoal line, indicating that they have the similar performance on both writer and script identification.

4.3. Historical manuscript dating

Historical manuscript dating has been studied recently in [45, 46, 47, 48, 49, 50, 51], which is the problem of automatically determining the date information of historical documents based on their handwriting styles and provides an efficient tool for historians or paleographers. The main challenge is how to extract the evolution of the handwriting styles over time. In this section, we provide the performance of different features which could capture the handwriting style on the historical manuscript dating problem on the Medieval Paleographical Scale

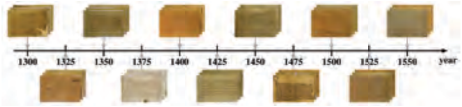


Figure 17: The time line of considered years on the MPS data set. Each document is labeled with one of the 11 key years.

(MPS) data set [52, 30, 51, 53] [1]. The MPS data set consists of 2858 images of charters produced between 1300-1550BC in four corners of old Dutch language area: Arnhem, Leiden, Leuven and Groningen [2]. Fig. 17 shows the time line of the MPS data set.

4.3.1. Evaluation criterion

For the dating problem, two measurements are widely used to measure the performance: the Mean Absolute Error (MAE) and Cumulative Score (CS) [54], which are defined as:

$$\begin{cases} MAE = \sum_{i=1}^{N} |\overline{K(y_i)} - K(y_i)|/N \\ CS(\alpha) = N_{e\leq\alpha}/N \times 100\% \end{cases} \tag{15}$$

where $K(y_i)$ is the ground-truth of the input query document y_i and $\overline{K(y_i)}$ is the corresponding estimated key year, $|\cdot|$ is the absolute operator, N is the number of query documents and $N_{e\leq\alpha}$ is the number of test images on which the key year estimation $\overline{K(y_i)}$ makes an absolute error $e = |\overline{K(y_i)} - K(y_i)|$ no higher than the acceptable error level: α years. For historians or paleographers, an error of ±25 is, more often than not, acceptable when dating the medieval historical documents on the MPS data set. Therefore, the error level α is set to 25 years in this section.

[1] The MPS data set has been collected by Petros Samara and Prof. Jan Burgers who study paleography.

[2] The project website is: `http://application02.target.rug.nl/monk/Projects/MPS/`

4.3.2. Dating by writer identity

The writers of 1127 documents are labeled, produced by 143 writers with at least two samples on the MPS data set formed the MPS-Writer Known with Multiple samples (MPS-WKM) subset, and the writers of 899 documents are unknown, formed the MPS-Writer Unknown(MPS-WU) subset. Following our previous work [30], we perform the writer identification on the MPS-WKM set and perform the dating using the K nearest neighbors (KNN) method on the MPS-WU data set, considering the rest of documents as training samples, whose writers are known.

Table 6 shows the results of writer identification and dating by writer identity on the MPS data set with different features. From Table 6 we can see that the CoHinge and QuadHinge provide the best results for writer identification and Junclets gives the best results among the grapheme-based features. For dating by KNN, the best performance is achieved by Junclets when K≤10 and by CO^3 when K>10. For each feature, the performance of dating decreases when K increases. Among the textural-based features, CoHinge gives the best performance when K=5 and QuadHinge provides the best results when K>5. Fig. 18 shows the relationship of Top-10 performance of different features for writer identification and dating by KNN with K=10 on the MPS data set. From the figure we can see that the CoHinge and QuadHinge features are discriminative both for writer identification and dating.

4.3.3. Dating by general handwriting style classification

In this section, we conduct historical manuscript dating by general handwriting style classification, in which all documents from each key year are considered as a class (there is an obvious border between nearby key years in the MPS data set) and a linear Support Vector Machine (SVM) is trained to predict the date information, following the work [30, 55]. All the documents on the MPS data set are randomly divided into two parts: a training set (70%) and a testing set (30%). The experiments are repeated 20 times and the average results with standard deviations are reported in this section.

Table 6: The performance of writer identification and dating by writer identify in terms of MAEs and CS($\alpha = 25$) of different features on the MPS data set using nearest neighbor method.

Feature	Writer identification		Dating by hit-document key year (KNN)							
	143 writers		1959 training and 899 query samples							
	1127 documents		K=5		K=10		K=20		K=50	
	Top-1	Top-10	MAEs	CS($\alpha = 25$)	MAEs	CS($\alpha = 25$)	MAEs	CS($\alpha = 25$)	MAEs	CS($\alpha = 25$)
LBP	36.0	68.7	54.1	49.4%	50.1	51.7%	50.3	50.0%	50.0	49.8%
CoLBP	50.5	77.3	39.2	61.2%	39.2	60.0%	39.4	60.1%	42.8	55.6%
RLH	54.4	78.8	33.8	66.8%	32.3	66.2%	33.2	65.1%	34.2	62.1%
LBPruns	64.8	83.0	27.3	72.5%	27.8	70.1%	27.6	69.7%	30.3	66.9%
Hinge	67.9	83.1	27.2	71.6%	27.9	71.5%	28.6	69.1%	33.5	63.6%
CoHinge	**76.5**	89.1	19.2	78.9%	21.8	76.0%	26.5	69.1%	32.3	64.1%
Δ^1Hinge	59.5	79.4	36.1	66.1%	37.1	64.2%	37.1	62.9%	40.7	58.2%
TCC	70.8	86.1	25.5	71.8%	27.3	68.8%	29.8	66.5%	37.0	60.5%
QuadHinge	75.8	**89.4**	20.4	78.6%	21.6	75.8%	25.0	72.4%	30.8	66.0%
Quill	65.0	83.1	34.1	67.1%	36.1	65.3%	38.9	62.7%	43.9	57.4%
QuillHinge	65.3	85.3	25.5	74.4%	26.9	71.9%	26.4	70.4%	31.5	65.2%
COLD	70.6	85.3	25.9	71.8%	29.0	67.8%	31.1	66.1%	39.9	57.8%
CO^3	65.5	82.5	24.5	75.2%	23.1	76.4%	**23.4**	**75.2**%	**27.8**	70.2%
kCF	64.7	84.6	22.9	76.4%	24.1	74.2%	24.5	73.1%	26.7	**70.8**%
Junclets	72.9	87.6	**18.4**	**81.7**%	**20.6**	**78.7**%	24.3	73.6%	29.4	67.4%
kSF	60.9	79.8	30.6	70.2%	30.3	67.6%	31.1	66.0%	35.5	61.1%
IC	58.7	80.0	29.5	72.5%	29.2	71.4%	31.0	69.1%	37.2	62.5%

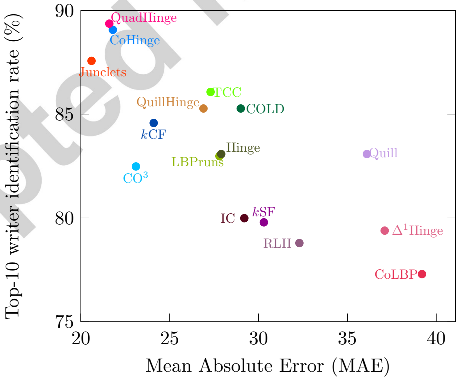


Figure 18: The Top-10 performance of writer identification with respect to the dating by KNN with K=10. Note that the features closed to the top-left corner are discriminative both for writer identification and dating on the MPS data set. (The LBP is not on the figure because its low writer identification performance (<75%))

Two different evaluation scenarios are considered, depending on whether including the writer duplicates on the training and testing data sets [50]:

1. *wr.excl.* scenario: documents from the same writer are carefully selected and they should be only in the training or testing sets, never appear in both training and testing sets.

2. *wr.incl.* scenario: documents are randomly splitting into training and testing sets, without the consideration of the writer labels.

Table 7 shows the results of different features on the MPS data set. The best three results are achieved by CoHinge, QuadHinge and Junclets, which are much better than the Hinge and Quill features. Table 8 gives the performance of the feature combination between the texture-based and grapheme-based features. We can observe that the performance of feature combination is not necessary better than the best result of the individual features involved in the combination. For the informative Junclets feature, only combining with Hinge, CoHinge and TCC gives an improvement in the scenario of *wr.incl.* Combining different grapheme-based features with QuadHinge provides a worse result, except the kSF feature in the *wr.incl.* scenario.

Table 7: The MAEs and CS($\alpha = 25$)s of the historical manuscript dating using SVM on the MPS data set. The best three performance are highlighted.

Feature	*wr.excl.* scenario		*wr.incl.* scenario	
	MAEs	CS($\alpha = 25$)	MAEs	CS($\alpha = 25$)
LBP	39.9±4.9	59.3±6.3%	34.3±8.1	65.2±7.7%
CoLBP	48.0±16.1	53.8±13.1%	41.2±19.7	59.6±18.3%
RLH	39.9±3.5	61.6±3.7%	31.4±1.2	68.3±2.1%
LBPruns	26.2±5.4	77.5±5.2%	17.0±2.9	84.3±3.5%
Hinge	23.6±3.3	77.9±3.2%	13.4±0.7	87.4±1.1%
CoHinge	16.6±3.0	85.9±3.7%	7.4±0.6	93.3±1.1%
Δ^1Hinge	32.2±3.6	69.1±4.3%	20.5±0.9	80.7±1.4%
TCC	18.2±2.5	83.9±3.3%	9.7±0.8	90.8±1.3%
QuadHinge	**14.0**±1.6	**89.5**±2.5%	**6.4**±0.5	**94.8**±1.0%
Quill	27.6±2.8	75.5±3.1%	16.6±1.2	84.3±1.9%
QuillHinge	20.5±2.9	82.4±3.3%	10.3±0.6	90.7±1.1%
COLD	23.8±2.9	77.6±3.7%	13.4±1.1	87.1±1.7%
CO^3	20.3±2.9	82.1±3.2%	11.5±0.8	89.5±1.5%
kCF	24.0±3.1	79.3±3.4%	14.7±1.1	86.6±1.7%
Junclets	14.5±1.7	89.3±1.8%	7.4±0.4	94.1±1.1%
kSF	18.7±2.6	84.4±3.1%	9.6±1.0	91.7±1.2%
IC	21.7±2.4	80.9±3.0%	12.5±0.8	88.2±1.7%

Table 8: The performance of historical manuscript dating based on feature combination on the MPS data set. The MAEs with red color decreases while with blue color increases compared to the best result of the individual features involved in the combination. Note that the smaller the MAE, the better the performance achieved.

Feature2 / Feature1	CO^3		kCF		Junclets		kSF		IC		Average	
	wr.excl.	*wr.incl.*	*wr.excl.*	*wr.incl.*	*wr.excl.*	*wr.incl.*	*wr.excl.*	*wr.incl.*	*wr.excl.*	*wr.incl.*	*wr.excl.*	*wr.incl.*
LBP	18.8±2.3	10.7±0.9	22.7±3.0	13.2±0.9	15.6±2.6	7.5±0.6	20.0±2.7	9.8±1.3	20.7±2.4	11.4±0.8	19.56	10.52
CoLBP	20.3±3.0	10.9±0.8	23.1±5.9	16.0±4.1	16.8±2.4	8.7±1.0	20.4±2.8	12.4±2.2	19.9±3.0	11.4±1.4	20.10	11.88
RLH	21.9±3.7	12.3±1.0	23.8±3.4	15.1±1.1	17.6±2.4	9.1±0.9	23.0±2.6	13.4±1.1	22.1±3.5	13.4±0.9	21.68	12.66
LBPruns	25.8±3.9	15.1±2.4	26.4±6.4	16.4±4.8	28.7±7.9	15.6±2.6	29.6±6.7	17.3±4.8	25.3±3.8	14.9±2.0	29.16	15.86
Hinge	16.5±2.3	8.9±0.8	19.0±2.4	11.1±0.9	15.7±1.5	7.0±0.7	16.3±3.3	8.4±0.6	18.3±1.9	9.9±1.1	17.16	9.06
CoHinge	15.9±2.0	7.9±0.6	17.0±2.2	8.5±0.8	14.8±2.2	6.8±0.6	14.5±2.3	7.0±0.7	16.2±1.8	8.4±0.5	15.68	7.72
Δ^1Hinge	18.9±2.1	10.2±0.8	22.2±3.2	12.8±0.8	15.8±2.5	7.5±0.6	18.4±2.5	9.1±0.6	19.0±3.2	10.1±0.9	18.86	9.94
TCC	16.7±2.8	8.9±0.7	17.7±2.4	9.7±0.8	14.9±2.3	6.9±0.5	14.3±2.0	7.9±0.7	16.9±2.9	8.6±0.8	16.10	8.40
QuadHinge	15.9±2.4	8.7±0.7	16.7±2.0	8.6±0.6	**13.6**±2.2	**6.6**±0.5	13.7±2.1	6.9±0.7	16.9±2.8	8.6±0.9	15.30	7.88
Quill	18.1±2.1	10.1±0.7	21.7±2.6	12.5±0.9	14.8±2.7	7.6±0.9	17.3±1.9	9.2±0.8	18.5±1.9	10.9±0.9	18.08	10.06
QuillHinge	19.1±2.5	10.2±0.5	21.2±2.5	12.0±0.8	15.1±2.3	7.5±0.6	17.2±2.7	8.7±0.8	19.6±2.1	11.0±0.9	18.44	9.88
COLD	19.8±2.9	9.9±0.8	19.9±2.3	11.0±0.8	17.9±2.3	8.8±0.7	19.5±2.3	10.4±0.8	18.5±2.1	9.9±0.9	19.12	10.00
Average	18.97	10.32	20.95	12.24	16.78	8.30	18.68	10.01	19.33	7.99	-	-

4.4. Manuscript localization

In this section, we conduct the experiments of historical manuscript localization based on the MPS data set, where the historical manuscripts are from four cities: Arnhem, Leiden, Leuven and Groningen. The KNN and linear SVM are used to evaluate the performance of the features for manuscript localization.

Fig. 19 presents the results of different features with respect to the number of K using the KNN method. The best result is achieved by the LBP feature and the performance of the CoHinge and QuadHinge features are comparable, which are better other features, except LBP. Fig. 20 shows the performance using the linear SVM classification. The QuadHinge reaches the recognition rate 94.0% and CoHinge reaches 92.8%, which are higher than other features, as well as their performance when using the KNN method.

5. Discussion and conclusion

In this paper, we have presented a joint feature distribution principle to demonstrate how to design novel and effective features based on the existed

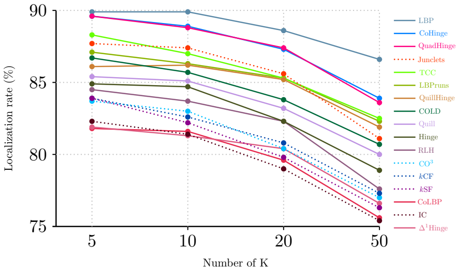


Figure 19: The localization rates of the features with respect to the number of K using the KNN method on the MPS data set. Note that the legend is on the descending order according to the performance when K=10.

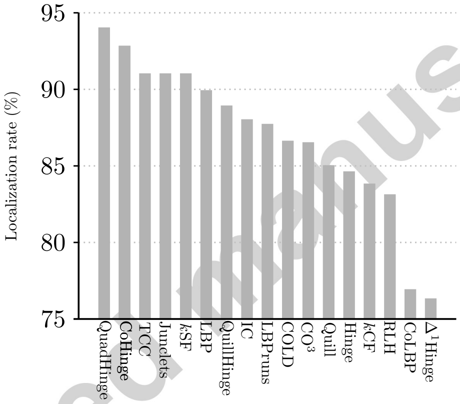


Figure 20: The localization rates with different features using the linear SVM classification.

feature methods. Seventeen features have been evaluated for handwritten document understanding beyond OCR. From the experimental results we can obtain the conclusions that: (1) The co-occurrence features are powerful than their original features for writer identification. For example, CoLBP provides better results than LBP and LBPruns gives better results than LBP and Run-lengths; (2) The proposed CoHinge and QuadHinge features provide the best results for writer identification on five data sets. The CoHinge and QuadHinge features are so discriminative that combining them with other grapheme-based features can not provide an improvement; (3) The best results of script identification are given by the contour fragments kCF and the QuadHinge features, because they can capture the character shape information, which takes an important role for script identification. The QuadHinge feature obtains the best results when doing the writer and script identification simultaneously. (4) The CoHinge, Junclets and QuadHinge are more powerful for historical manuscript dating and localization. However, LBP obtains the best performance for manuscript dating using the KNN method. From the experimental results, we can conclude that our novel QuadHinge and CoHinge features present the promising results for the four problems: writer and script identification, historical document dating and localization. In future work, more kernel functions could be investigated to achieve more powerful, as well as transform-invariant features.

Acknowledgments

This work has been supported by the Dutch Organization for Scientific Research NWO (project No. 380-50-006). The authors would like to thank Shijie Zhao and Yanfang Feng to collect the CERUG data set and Petros Samara and Prof.Jan Burgers to collect the MPS data set.

References

[1] P. A. Stokes, Digital approaches to paleography and book history: Some challenges, present and future, Frontiers in Digital Humanities 2 (2015) 5.

[2] M. Bulacu, L. Schomaker, Text-independent writer identification and verification using textural and allographic features, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (4) (2007) 701–717.

[3] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (10) (2005) 1615–1630.

[4] Y. Li, S. Wang, Q. Tian, X. Ding, Feature representation for statistical-learning-based object detection: A review, Pattern Recognition 48 (11) (2015) 3542–3559.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database, in: Conference on Computer Vision and Pattern Recognition (CVPR).

[6] X. Qi, R. Xiao, C.-G. Li, Y. Qiao, J. Guo, X. Tang, Pairwise rotation invariant co-occurrence local binary pattern, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (11) (2014) 2199–2213.

[7] S. He, M. Wiering, L. Schomaker, Junction detection in handwritten documents and its application to writer identification, Pattern Recognition 48 (12) (2015) 4036–4048.

[8] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4) (2002) 509–522.

[9] R. M. Haralick, K. Shanmugam, I. H. Dinstein, Textural features for image classification, IEEE Transactions on Systems, Man and Cybernetics (6) (1973) 610–621.

[10] S. Yang, M. Chen, D. Pomerleau, R. Sukthankar, Food recognition using statistics of pairwise local features, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2249–2256.

[11] S. Ito, S. Kubota, Object classification using heterogeneous co-occurrence features, in: Computer Vision–ECCV, 2010, pp. 701–714.

[12] A. Brink, J. Smit, M. Bulacu, L. Schomaker, Writer identification using directional ink-trace width measurements, Pattern Recognition 45 (1) (2012) 162–171.

[13] H. E. Said, T. N. Tan, K. D. Baker, Personal identification based on handwriting, Pattern Recognition 33 (1) (2000) 149–160.

[14] A. J. Newell, L. D. Griffin, Writer identification using oriented basic image features and the delta encoding, Pattern Recognition 47 (6) (2014) 2255–2265.

[15] S. He, L. Schomaker, Delta-n hinge: Rotation-invariant features for writer identification, in: International Conference on Pattern Recognition (ICPR), 2014, pp. 2023–2028.

[16] L. Fei-Fei, P. Perona, A bayesian hierarchical model for learning natural scene categories, in: Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, 2005, pp. 524–531.

[17] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (7) (2002) 971–987.

[18] G. Zhao, M. Pietikainen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 915–928.

[19] D. Bertolini, L. S. Oliveira, E. Justino, R. Sabourin, Texture-based descriptors for writer identification and verification, Expert Systems with Applications 40 (6) (2013) 2069–2080.

[20] Y. Hannad, I. Siddiqi, M. E. Y. El Kettani, Writer identification using texture descriptors of handwritten fragments, Expert Systems with Applications 47 (2016) 14–22.

[21] B. Arazi, Handwriting identification by means of run-length measurements, IEEE Trans. Syst., Man and Cybernetics (12) (1977) 878–881.

[22] C. Djeddi, I. Siddiqi, L. Souici-Meslati, A. Ennaji, Text-independent writer recognition using multi-script handwritten texts, Pattern Recognition Letters 34 (10) (2013) 1196–1202.

[23] A. Gordo, F. Perronnin, E. Valveny, Large-scale document image retrieval and classification with runlength histograms and binary embeddings, Pattern Recognition 46 (7) (2013) 1898–1905.

[24] S. He, L. Schomaker, General pattern run-length transform for writer identification, in: International Workshop on Document Analysis Systems (DAS), 2016, pp. 60–65.

[25] S. He, L. Schomaker, Co-occurrence features for writer identification, in: International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016.

[26] I. Siddiqi, N. Vincent, Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features, Pattern Recognition 43 (11) (2010) 3853–3865.

[27] S. Benhamou, How to reliably estimate the tortuosity of an animal's path:: straightness, sinuosity, or fractal dimension?, Journal of theoretical biology 229 (2) (2004) 209–220.

[28] D. K. Prasad, C. Quek, M. K. Leung, S.-Y. Cho, A parameter independent line fitting method, in: Asian Conference on Pattern Recognition (ACPR), 2011, pp. 441–445.

[29] L. Schomaker, M. Bulacu, Automatic writer identification using connected-component contours and edge-based features of uppercase western script, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (6) (2004) 787–798.

[30] S. He, P. Samara, J. Burgers, L. Schomaker, Image-based historical manuscript dating using contour and stroke fragments, Pattern Recognition 58 (2016) 159–171.

[31] X. Wang, B. Feng, X. Bai, W. Liu, L. J. Latecki, Bag of contour fragments for robust shape classification, Pattern Recognition 47 (6) (2014) 2116–2125.

[32] T. Kohonen, Self-organization and associative memory, Springer Verlag 1.

[33] L. Schomaker, L. Vuurpijl, Forensic writer identificaiton: a benchmark data set and a comparison of two systems, Technical Report, Nijmegen: NICI, 2000.

[34] U.-V. Marti, H. Bunke, The IAM-database: an English sentence database for offline handwriting recognition, International Journal on Document Analysis and Recognition 5 (1) (2002) 39–46.

[35] A. Hassaine, S. A. Maadeed, ICFHR 2012 competition on writer identification challenge 2: Arabic scripts, in: International Conference on Frontiers in Handwriting Recognition (ICFHR), 2012, pp. 835–840.

[36] G. Louloudis, B. Gatos, N. Stamatopoulos, A. Papandreou, ICDAR 2013 competition on writer identification, in: International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 1397–1401.

[37] F. Kleber, S. Fiel, M. Diem, R. Sablatnig, CVL-database: An off-line database for writer retrieval, writer identification and word spotting, in: International Conference on Document Analysis and Recognition (ICDAR), 2013, pp. 560–564.

[38] D. Ghosh, T. Dube, A. P. Shivaprasad, Script recognitiona review, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (12) (2010) 2142–2161.

[39] A. Busch, W. W. Boles, S. Sridharan, Texture for script identification, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (11) (2005) 1720–1732.

[40] A. L. Spitz, Determination of the script and language content of document images, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (3) (1997) 235–245.

[41] L. Shijian, C. L. Tan, Script and language identification in noisy and degraded document images, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (1) (2008) 14–24.

[42] B. Shi, X. Bai, C. Yao, Script identification in the wild via discriminative convolutional neural network, Pattern Recognition 52 (2016) 448–458.

[43] J. Hochberg, K. Bowers, M. Cannon, P. Kelly, Script and language identification for handwritten document images, International Journal on Document Analysis and Recognition 2 (2-3) (1999) 45–52.

[44] G. Zhu, X. Yu, Y. Li, D. Doermann, Language identification for handwritten document images using a shape codebook, pattern recognition 42 (12) (2009) 3184–3191.

[45] S. He, P. Samara, J. Burgers, L. Schomaker, Towards style-based dating of historical documents, in: International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014, pp. 265–270.

[46] F. Wahlberg, L. Mårtensson, A. Brun, Large scale style based dating of medieval manuscripts, in: Workshop on Historical Document Imaging and Processing (HIP), 2015, pp. 107–114.

[47] 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.

[48] Y. Li, D. Genzel, Y. Fujii, A. C.Popat, Publication date estimation for printed historical documents using convolutional neural networks, in: Workshop on Historical Document Imaging and Processing (HIP), 2015, pp. 99–106.

[49] N. R.Howe, A. Yang, M. Penn, A character style library for Syriac manuscripts, in: Workshop on Historical Document Imaging and Processing (HIP), 2015, pp. 123–128.

[50] S. He, P. Samara, J. Burgers, L. Schomaker, Historical document dating using unsupervised attribute learning, in: International Workshop on Document Analysis Systems (DAS), 2016, pp. 36–41.

[51] S. He, P. Samara, J. Burgers, L. Schomaker, Historical manuscript dating based on temporal pattern codebook, Computer Vision and Image Understanding, doi: 0.1016/j.cviu.2016.08.008.

[52] P. Samara, Towards a medieval palaeographical scale (1300-1550), in: Papsturkundenforschung zwischen internationaler Vernetzung und Digitalisierung, Munich, 2014.

[53] S. He, P. Samara, J. Burgers, L. Schomaker, A multiple-label guided clustering algorithm for historical document dating and localization, IEEE Transactions on Image Processing 25 (11) (2016) 5252–5265.

[54] X. Geng, Z.-H. Zhou, K. Smith-Miles, Automatic age estimation based on facial aging patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (12) (2007) 2234–2240.

[55] F. Palermo, J. Hays, A. A. Efros, Dating historical color images, in: Computer Vision–ECCV, 2012, pp. 499–512.