Extracting Maya Glyphs from Degraded Ancient Documents via Image Segmentation

RUI HU, Idiap Research Institute
JEAN-MARC ODOBEZ and DANIEL GATICA-PEREZ, Idiap Research Institute and École Polytechnique Fédérale de Lausanne (EPFL)

We present a system for automatically extracting hieroglyph strokes from images of degraded ancient Maya codices. Our system adopts a region-based image segmentation framework. Multi-resolution super-pixels are first extracted to represent each image. A Support Vector Machine (SVM) classifier is used to label each super-pixel region with a probability to belong to foreground glyph strokes. Pixelwise probability maps from multiple super-pixel resolution scales are then aggregated to cope with various stroke widths and background noise. A fully connected Conditional Random Field model is then applied to improve the labeling consistency. Segmentation results show that our system preserves delicate local details of the historic Maya glyphs with various stroke widths and also reduces background noise. As an application, we conduct retrieval experiments using the extracted binary images. Experimental results show that our automatically extracted glyph strokes achieve comparable retrieval results to those obtained using glyphs manually segmented by epigraphers in our team.

CCS Concepts: • **Computing methodologies** → **Image segmentation**; *Visual content-based indexing and retrieval*; *Image representations*; • **Applied computing** → *Arts and humanities;*

Additional Key Words and Phrases: Ancient maya codices, document binarization, hieroglyph retrieval

ACM Reference Format:

Rui Hu, Jean-Marc Odobez, and Daniel Gatica-Perez. 2017. Extracting maya glyphs from degraded ancient documents via image segmentation. ACM J. Comput. Cult. Herit. 10, 2, Article 10 (March 2017), 22 pages.
DOI: http://dx.doi.org/10.1145/2996859

1. INTRODUCTION

The ancient Maya language infused art with unique pictorial forms of hieroglyphic writing and has left us an exceptionally rich legacy of the ancient Maya civilization. Most Maya texts were produced during the classic period of the ancient Mayan civilization (AD 250–900), throughout which hieroglyphic texts were carved or painted on various media types, including stone monuments, personal items, and so on. A special type of Maya text was discovered written on screenfold books (the so-called

This work is supported by the Swiss National Science Foundation (SNSF) through the project MAAYA: Multimedia Analysis and Access for Documentation and Decipherment of Maya Epigraphy.

Authors' addresses: Idiap Research Institute, Centre du Parc, Rue Marconi 19, 1920 Martigny, Switzerland; R. Hu, J.-M. Odobez, and D. Gatica-Perez; emails: {rhu, odobez, gatica}@idiap.ch.

DOI: http://dx.doi.org/10.1145/2996859

Maya codices), made of bark cloth. Only three such books are known to have survived the Spanish Conquest (namely the Dresden, Madrid, and Paris codexes, respectively). These books were produced in all likelihood within the greater peninsula of Yucatan at some point during the post-classic period (AD 1000–1519).

Similarly to many cultural heritage documents, Maya scripts from the surviving ancient codices often lose their visual quality over time due to degradation and staining. In addition to the fading of ink, background noise is often introduced due to ink stains, deterioration of the cloth material, and so on. Therefore, extracting clean glyph strokes from raw images of the ancient scripts is a challenging yet crucial step for any further automatic document image analysis tasks such as page layout analysis, character recognition, and so on. Previous research on automatic Maya hieroglyph analysis assume that high-quality glyph images are available. In order to produce such data, epigraphers spend a considerable amount of time manually extracting glyph strokes from noisy background [Hu et al. 2015]. This process cannot be scaled over large datasets, due to the amount of effort required from experts.

Document image binarization aims to separate the foreground texts from the document background. It is a key step in all document image processing systems, the accuracy of which will largely affect the performance of any following tasks. Despite the fact that image binarization has been addressed in the literature in the past few decades, segmenting text strokes from badly degraded historic document images is still a challenging task, due to high inter-/intra-variation between foreground text strokes and the document background, as well as the different degradation types/levels across document images. Various binarization methods have been evaluated for different types of documents, such as ancient English, French, or Greek manuscripts, and so on [Fischer et al. 2014]. These documents, however, have rather different appearance features compared to ancient Maya scripts. Ancient Egyptian hieroglyphs [Franken and van Gemert 2013] and Chinese characters [Lu et al. 2013] are perhaps the most similar to our data. To the best of our knowledge, the automatic extraction of Maya hieroglyph strokes from degraded noisy document images has not been studied.

Figure 1 shows examples of glyph block images cropped from the three surviving ancient Maya codices (Figure 1(a)); their clean raster versions (Figure 1(b)), and high-quality reconstructed vectorial images (Figure 1(c)), generated by epigraphers in our team. Experts can spend hours in this process. The clean raster images are produced by manually removing background areas from the raw image, whereas the reconstructed forms are generated by further carefully reconstructing the broken lines and missing strokes. Heavily damaged glyphs are reconstructed by experts through comparative analysis to semantically understand the glyph through its remaining strokes as well as its context information. Although the reconstructed images represent higher-quality hieroglyphs, the clean raster forms preserve details that are more close to the original painting, which could represent a higher historic value.

The objective in this article is to propose a fully automatic system capable of extracting glyph strokes as close as possible to the clean raster forms (Figure 1(b)) from the raw images (Figure 1(a)). We use the binarized forms (Figure 1(d)) of the clean raster images as our ground truth. Our objective is to provide a tool that supports epigraphers with a time-consuming task. From Figure 1, we can see that the challenges of our data include the following. First, glyphs show significant variations in terms of stroke width and brightness. Second, various degradations appear in both the glyph strokes and the background, which results in broken lines and missing strokes, as well as background roughness. Third, there are different contrasts between foreground and background, both across the image dataset and within individual images. Last but not least, cropped glyph block images often contain extra strokes from neighboring blocks and bands/lines used to separate pages in the original codices.

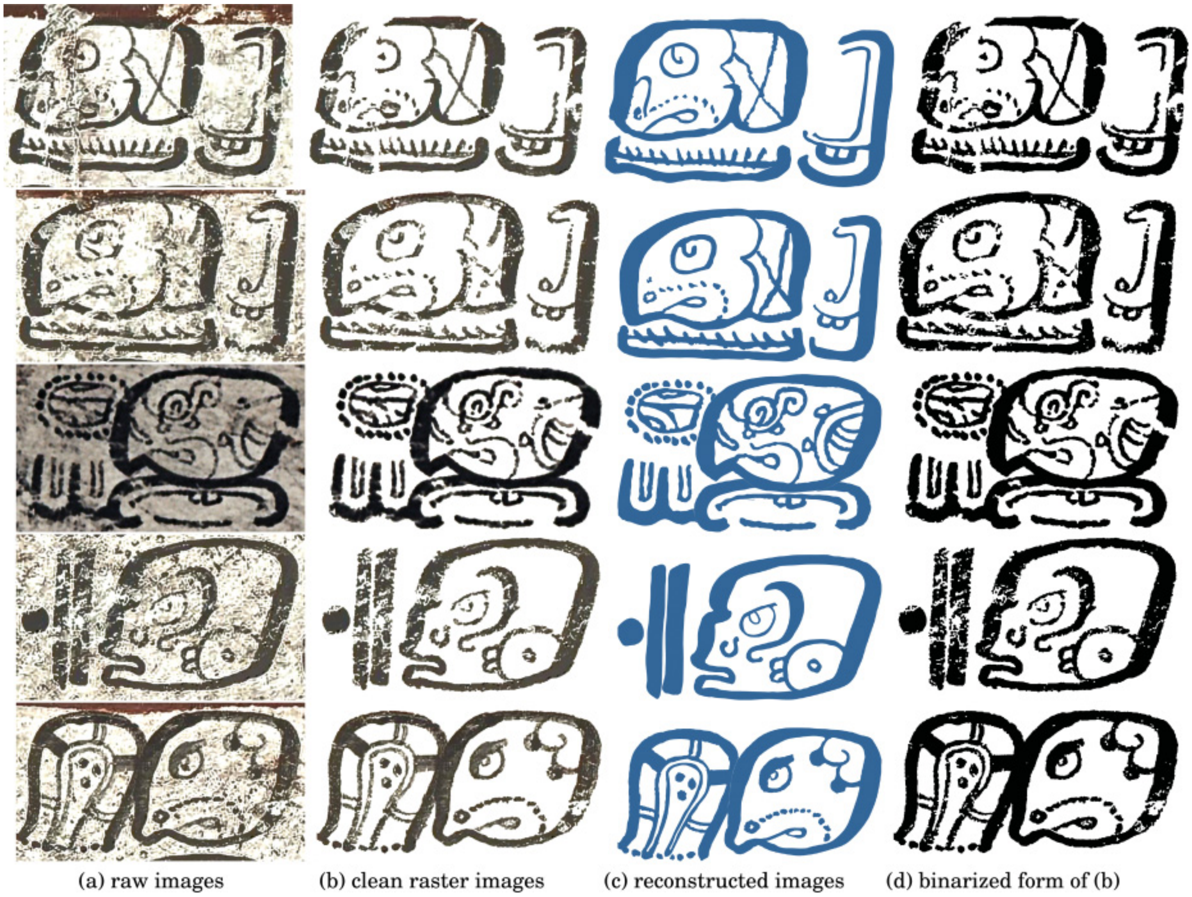


Fig. 1. (a) Examples of glyph blocks; (b) manually generated clean raster images; (c) reconstructed high-quality vectorial images; (d) binarized forms of the clean raster images. Images (a)-(c) were manually cropped and produced by Carlos Pallán Gayol (supported by the German Research Foundation through the MAAYA project, University of Bonn, used with permission).

Our contributions include the following:

—First, we present a system that automatically extracts glyph strokes from noisy images of degraded ancient Maya codices. Our system achieves promising pixelwise labeling results, compared with the ground-truth images manually segmented by epigraphers in our team.

—Second, the resulting binary images are further tested in a glyph retrieval system, the performance of which is comparable to that achieved using manually generated clean raster images. This result is particularly encouraging, because it justifies the use of our system for the processing of large datasets of degraded ancient documents, which will significantly reduce the time and effort spent by epigraphers on manual data preparation.

The presented method first adopts a region-based image segmentation framework to extract multi-resolution super-pixel regions. A SVM classifier is then applied to label each region with a probability to belong to foreground glyph strokes. Pixelwise probability maps from multiple resolution regions are aggregated. Finally, a fully connected Conditional Random Field (CRF) model is incorporated to improve the labeling consistency. Our proposed method can in principle be used in any document image binarization task, especially cases where simple thresholding methods do not work well, like eroded ancient documents with various stroke widths and details.

The rest of the article is organized as follows. Related works in document image binarization and image segmentation are reviewed in Section 2. Section 3 briefly introduces our data sources (ancient Maya codices) and the data processing conventions epigraphers in our team followed to produce our data. Our method is introduced in Section 4. Experimental settings, results, and discussion are given in Section 5. Concluding remarks are presented in the last section.

2. RELATED WORK

The digitization of historic documents creates the opportunity to access and study a huge amount of cultural heritage resources. Automatic document analysis is a crucial step for efficient search and browsing of ancient manuscripts in digital libraries. Various image processing, machine learning, information retrieval, and data management techniques have been applied to assist this task.

Image restoration, page layout analysis, character recognition, and semantic document understanding are the typical research questions for automatic document image analysis. Due to the fact that historic documents are often degraded, it is challenging to apply character analysis algorithms directly on the raw images. Therefore, automatically extracting foreground text from noisy background becomes an important pre-processing step, which can be considered an image binarization problem.

In the following, we first review the main trends of the image segmentation literature in general and then discuss in more detail techniques that are more specifically targeted towards document image binarization.

2.1 Image Segmentation

Image segmentation aims to divide an image into meaningful (semantic object) or local homogeneous regions sharing similar attributes. Image segmentation systems can be classified into two categories: interactive and fully automatic systems. Interactive systems utilize user input to provide coarse or partial annotations as prior knowledge and have been applied in many commercial products. In such systems, users provide object information by marking object boundaries [Gleicher 1995], placing a bounding box around [Rother et al. 2004] or loosely drawing scribble on object regions [Boykov and Jolly 2001]. Fully automatic segmentation systems include unsupervised and supervised methods. Unsupervised methods segment an image into homogeneous regions based on appearance consistency using clustering algorithms such as *k*-means and Gaussian mixture model, model shifting [Comaniciu and Meer 2002], contour-based methods [Arbelaez et al. 2011], graph partitioning [Felzenszwalb and Huttenlocher 2004; Shi and Malik 2000], region merging and splitting methods [Vincent and Soille 1991], and so on. Supervised learning methods are often used to produce pixelwise labeling according to a pre-defined set of categories, such as the semantic image segmentation task. Given the complex nature of our data and the amount of data we would like to process, in this article, we are interested in automatic image segmentation with supervised learning methods.

State-of-the-art supervised image segmentation methods typically consist of two main components: local object appearance and local consistency of neighboring pixels or image regions. These components are generally integrated into a unified probabilistic framework, such as a CRF [Kohli et al. 2009], or are learned independently and used at different stages of a sequential framework [Csurka and Perronnin 2011].

Local appearance is often described by Gaussian derivative filter outputs, color, and location computed as pixel-level image features [Shotton et al. 2008]; Scale Invariant Feature Transform (SIFT) and color statistics computed on patches that are extracted either on a grid [Csurka and Perronnin 2011] or at detected interest point locations [Yang et al. 2007]; or color, texture, and shape features computed on image regions [Hu et al. 2012]. These low-level features can also be used to build higher-level representations such as semantic texton forests [Shotton et al. 2008], bag-of-visual-words [Verbeek and Triggs 2007], or Fisher vectors [Csurka and Perronnin 2011]. Extracted features are often fed into a classifier that predicts class labels at the pixel level [Wang et al. 2012], patch level [Csurka and Perronnin 2011], or region level [Yang et al. 2007].

Local consistency is generally enforced via pairwise constraints between neighboring pixels and can be based on a simple Potts model that penalizes equally all class transitions between neighbor pixels [Verbeek and Triggs 2007] or can be contrast sensitive where the penalty depends also on the pixel values [Krähenbühl and Koltun 2012]. To enforce region-level consistency, higher-order potentials can be added to the CRF model [Ladicky et al. 2009], or, alternatively, hard constrains ensure that all pixels within a low-level region have the same label. The latter can be done as a post-processing step, that is, averaging the class likelihood of each pixel within a region or describing directly the regions and learning to predict class labels at a region level [Yang et al. 2007]. Recently, a fully connected CRF model [Krähenbühl and Koltun 2012] has been used in image segmentation to establish pairwise potentials on all pairs of pixels in the image and has been shown to achieve promising results. In this article, we propose to explore this fully connected CRF model in our glyph stroke extraction framework to improve pixelwise labeling consistency.

2.1.1 *Document Image Binarization.* Document image binarization aims to automatically extract foreground text from a noisy background. While it can be considered a direct extension of the image segmentation task, it has also been a research topic on its own, due to specificities of the problem and documents to be handled. Below we review some of the most popular methods.

Thresholding-based algorithms are among the most popularly used in historic document image binarization tasks and produce state-of-the-art performance [Pratikakis et al. 2013; Su et al. 2013]. Simple thresholding methods include histogram-shape-based thresholding [Rosenfeld and de la Torre 1983], where the histogram is forced into a two-peaked representation. Otsu's algorithm [Otsu 1979] is one of the most widely used global thresholding methods. It automatically selects an optimal threshold relying on a discriminant criterion minimizing the intra-class variance compared to the between-class variance. However, such methods fail in images with non-uniformly distributed intensity. Local adaptive thresholding methods provide an approach to deal with such images. A threshold is estimated at each pixel based on local statistics such as contrast [Bernsen 1986], variance [Niblack 1990], or mean and standard deviation [Sauvola and Pietikäinen 2000] within a local window around the pixel. The size of the local window should be large enough to cover sufficient foreground and background pixels but small enough to take into account nonuniform background distribution over the image. It is challenging to use either global or local thresholding algorithms on our data, since our documents are highly eroded with non-uniform background noise and contain strokes with various scales, widths, and details.

Approximate background subtraction has been conducted in many document image binarization frameworks to remove possible stains and general document background. In Lu et al. [2010], a document background model is first estimated through an interactive polynomial smoothing procedure. This method has achieved the top performance in the Document Image Binarization Contest 2009 [Gatos et al. 2009]. In Moghaddam and Cheriet [2012], the estimated background is calculated using a multi-scale approach, starting from a rough binarized initialization, and improved in a bootstrap process using an adaptive form of Otsu's method [Otsu 1979]. In Mitianoudis and Papamarkos [2015], an approximate background model is obtained by performing low-pass filtering. However, such methods struggle with highly noisy backgrounds and non-uniform pixel distributions.

Hybrid systems, which combine multiple binarization methods together, have also attracted a great deal of interest and provided promising results. The goal of such methods is to improve the output based on the assumption that different methods complement one another. In Gatos et al. [2008], a voting based approach is used to combine results from several binarization methods. The advantages of both global and local thresholding methods are combined in AL-Khatatneh et al. [2015] by automatically selecting the threshold value based on the comparison between global and local standard deviation. In Su et al. [2011], a classification framework is used to combine multiple thresholding methods by iteratively classifying uncertain pixels into foreground and background sets. Our method applies a similar strategy by combining multiple super-pixel resolutions to alleviate the need for selecting the resolution scale that should not only smooth the background noise but also preserve various stroke details.

3. DATA SOURCES AND PREPARATION

In this section, we first introduce the three surviving ancient Maya codices, from which our image data come from. We then briefly explain our data preparation steps and discuss data quality.

3.1 Data Sources

Our data come from three extant ancient Maya codices. The Dresden codex is housed at the Saxon State and University Library Dresden (SLUB), Germany.[1] In this work, our original image sources of the Dresden codex are from Förstemann [1880]. The Madrid codex is stored at the Museo de América in Madrid, Spain. The Paris codex resides at the Bibliothèque Nationale de France.[2] The Paris codex is the shortest among the three and consists of 24 pages of Maya scripts, whereas the Dresden and Madrid codices are longer (74 and 112 pages, respectively). While the exact provenance and dating of the Maya codices remains uncertain, most contemporary scholars consider that they were made within the northern Yucatan peninsula during the post-classic period of the ancient Maya culture. Given the inherent difficulties in the direct study and examination of the original hieroglyphic codex materials, our data sources consists of digital photographs or scans and online resources of the three codices as described in Vail and Hernández [2013].

3.2 Data Preparation and Groundtruth Generation

In a Maya codex, textual and icon information were painted using a brush on bark paper, coated with a film of lime plaster. Ancient Maya scribes usually divided codex pages into smaller sections using red bands/lines, which are referred to as *t'ols* by modern scholars, each *t'ol* being further divided into frames relevant to the specific dates, texts, and imagery depicted. Frames contain text areas of main signs, calendric glyphs, captions, and icons. A detailed template of an example *t'ol* (register) from the Dresden codex, "segmented" into its main elements (i.e., main glyph sign texts, captions, calendric signs, and icons) is shown in Hu et al. [2015]. In this article, we are interested in the main signs, which are typically composed of glyph blocks arranged in a gridlike pattern. The most common reading order of glyph blocks is from left to right and from top to bottom within double columns. Ancient Maya scribes followed complex writing conventions to render a particular term, and multiple glyphs are typically organized in a complex manner to form a block (see individual glyphs in Figure 2 that formed the first and the third blocks in Figure 1). Each individual glyph represents a semantic meaning (logogram) or a sound (syllable).

Epigraphers in our team first segment a codex page into main elements and then manually crop each main sign block with a bounding box (see Figure 1(a)), which we use as input raw images in our system. In order to provide pixelwise segmentation ground truth to evaluate our result, epigraphers manually

[1] High-resolution images of the Dresden codex are available at http://digital.slub-dresden.de/werkansicht/dlf/2967/1/.

[2] Paris codex: http://gallica.bnf.fr/ark:/12148/btv1b8446947j/f1.zoom.r=Codex%20Peresianus.langDE.

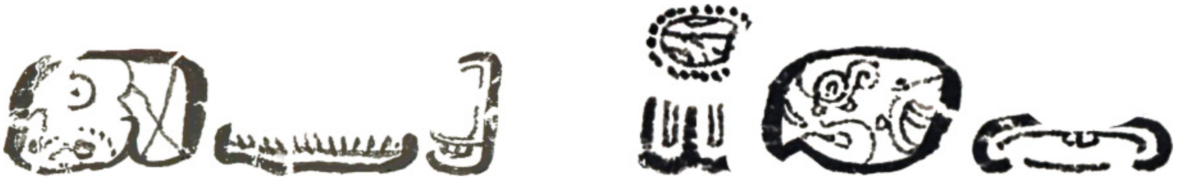


Fig. 2. Individual glyphs segmented from the first and the third blocks in Figure 1, respectively (images by Carlos pallán Gayol, supported by the German Research Foundation through the MAAYA project, University of Bonn, used with permission).

generated clean-raster images (Figure 1(b)) by separating the brush-strokes from background noise and preservation accidents. It requires epigraphers about 30min to generate a clean raster block, depending on the complexity and preservation factors of the original data.

Extracting and recognizing individual glyphs from a block is crucial for deciphering unknown Maya scripts. Epigraphers in our team manually segmented individual glyphs from blocks (see Figure 2). Individual glyphs are then annotated with their reading order within the block and are labeled based on four glyph catalogs spanning over 50 years of scholarly work [Thompson 1962; Macri and Vail 1901; Evrenov et al. 1961; Zimmermann 1956]. We take advantage of these annotations to use the automatically generated binary images in a glyph retrieval framework (see Section 5.3) to further evaluate our proposed glyph stroke extraction system.

3.3 Glyph Quality Analysis

In order to study data with different degradation levels, epigraphers in our team proposed a data quality ranking scheme. Each glyph is ranked with a scale from 0 to 4, representing glyph quality from lowest to highest. When a glyph quality is ranked "0," it means that its visual details are completely lost, it is not reconstructable, and its meaning is therefore not recoverable. A "1" means that the glyph has very poor visual quality, it is heavily mutilated and mostly not reconstructable, and therefore its meaning is usually not recoverable. A "2" indicates that the glyph stroke is partially missing; in some cases it is still recognizable through the remaining strokes and context. Glyphs with quality ranking "3" are in overall good condition, whereby some strokes are missing but still recognizable. Well-preserved glyphs are ranked "4." The percentage of glyphs, throughout the whole codices, with different quality rankings from 0 to 4 are 5%, 5%, 10%, 32%, and 48%, respectively.

4. GLYPH STROKE EXTRACTION VIA SEGMENTATION

We now introduce our region-based image segmentation system for extracting historic hieroglyph strokes from ancient Maya codices.

4.1 Super-Pixel Extraction

Super-pixels are usually over-segmented homogeneous regions, typically smaller than objects. The use of over-segmentation as a first step towards region-based image segmentation is a classic approach in image processing and computer vision [Roerdink and Meijster 2000]. Compared to pixel-based methods, the main benefits of using super-pixels include the smaller number of regions compared to pixels, which greatly reduces the model complexity, and the relative spatial support within each region, which increases the pixelwise labeling consistency within regions. Due to the fact that large amounts of ink stain "noise" often exist in the background of our image data, pixel-based methods could produce highly noisy segmentation results. In addition, region-based methods usually preserve object boundaries better than patch-based methods. Therefore, we adopt a super-pixel-based framework to produce binary segmentation results that preserve delicate stroke details and also smooth background noise.

Super-pixels can be generated by existing unsupervised low-level segmentation methods such as mean shift, contour detection [Arbelaez et al. 2011], and so on, by increasing the cluster number or controlling the region size. In this article, we adopt an efficient super-pixel extraction method: the Simple Linear Iterative Clustering (SLIC) [Achanta et al. 2012] algorithm.

SLIC clusters image pixels into local compact, homogeneous, and edge-aware regions using k-means. The algorithm contains three main steps: cluster center initialization, pixel label assignment, and cluster center updating. It begins by initializing k regularly spaced cluster centers on the image plane. The centers are then moved to the lowest gradient position in a 3×3 neighborhood to avoid edges. In the assignment step, each pixel i in the image plane is associated with its nearest cluster center. Note that SLIC only searches a $2S \times 2S$ region around each cluster center, where $S = \sqrt{N/K}$ represents the sampling interval of cluster centers and N is the number of pixels. This process leads to a significant speed advantage over traditional k-means where each pixel is compared with all cluster centers. The distance between pixel i and cluster center k combines color and location information according to

$$D = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2}, \tag{1}$$

with $d_c = \sqrt{(l_i - l_k)^2 + (a_i - a_k)^2 + (b_i - b_k)^2}$ and $d_s = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}$, where l, a, and b are the color values in the CIELab space, (x, y) represents the pixel position on the two-dimensional (2D) image plane, and m is the maximum color distance within a cluster. The m and S are normalization factors used to obtain a good compromise between the color and spatial distances whatever the number of sampled clusters or the image representation and contrast. In the update step, the cluster centers are adjusted to the mean vector $[l, a, b, x, y]$ of all pixels belonging to the cluster. The assignment and update steps are repeated iteratively until the the cluster center positions are stabilized.

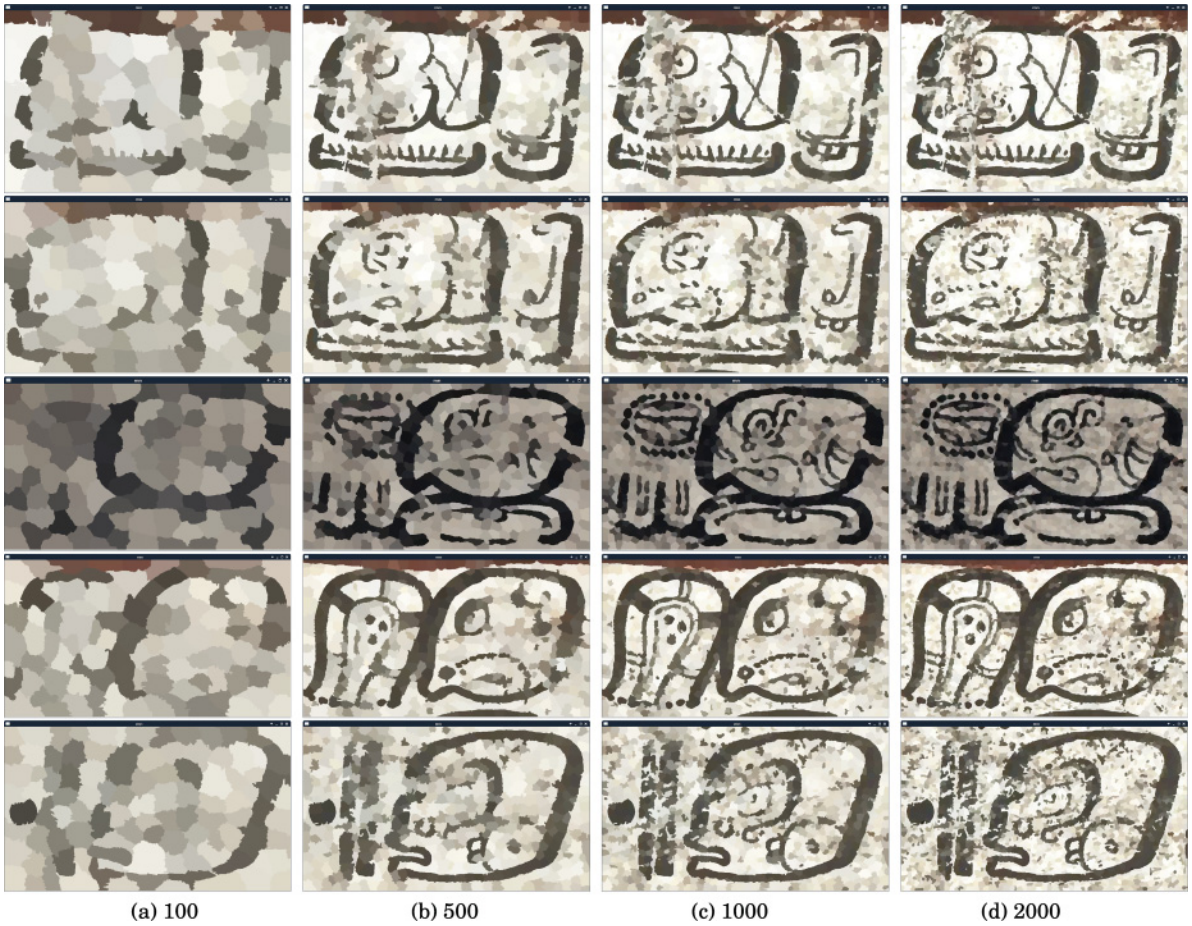
Similarly to Perazzi et al. [2012], we use an adaptation of SLIC super-pixels [Arbelaez et al. 2011]. In the adapted version, geodesic image distance [Criminisi et al. 2010] in CIELab space is used in the k-means algorithm, instead of Equation (1). The geodesic distance between pixel i and j is defined as $G = \min_{P \in \Gamma} d(P)$, where Γ is the set of all paths between i and j, and $d(P)$ is the cost associated to path P, which takes into account both the spatial distance and the image gradient information.

The algorithm requires specification of the number of clusters (super-pixels) to obtain. A larger number will produce a finer segmentation, whereas a smaller number leads to a coarser partition of the image. Figure 3 illustrates segmentation results with different number of super-pixels.

4.2 Pixelwise Labeling

To be able to obtain a map that encodes for each pixel the probability of belonging to foreground strokes, we learn SVM classifiers at multiple segmentation scales (i.e., by varying the number of super-pixels) and then aggregate class predictions over multiple scales.

4.2.1 *Super-Pixel Classification.* Each super-pixel is represented by its Local Color Statistic (LCS) and relative location information of the region centroid. LCS features are concatenated means and standard deviations in the R, G, and B channels computed within each region. The reason we use this simple color feature is that each region contains pixels of visually homogeneous colors, and color feature encodes important information to distinguish foreground text stroke from background area. A similar strategy has been used in the past [Salembier et al. 1998; Hu et al. 2012]. Additionally, in order to discriminate unwanted strokes from the meaningful text strokes, we incorporate the relative location information in the feature. Indeed, unwanted strokes are often from neighboring blocks or bands/lines used to divide codex pages and are usually located close to the boundary. The resulting feature is an eight-dimensional vector: $(I_R^m, I_G^m, I_B^m, I_R^s, I_G^s, I_B^s, x, y)$, where $I_{R,G,B}^m$ represent the average



(a) 100 (b) 500 (c) 1000 (d) 2000

Fig. 3. Visual comparison of images segmented into different predefined numbers of super-pixels. Each super-pixel is displayed using the average color of pixels within that region.

of pixel values within the super-pixel region in the R, G, B color channels individually, and $I^{s}_{R,G,B}$ are the corresponding standard deviations, and (x, y) refers to the relative spatial information of the region centroid.

In order to train classifiers, we need to label each region as either foreground or background. We first generate the ground-truth binary images (Figure 1(d)) by applying a simple thresholding on the clean raster images, followed with mathematical morphology operations to improve image quality by removing isolated pixels and closing degraded holes on glyph strokes. To label each super-pixel region, we use the pixel-level ground-truth masks as follows. We consider a super-pixel as a positive example if it has at least 50% of its area overlapping with the foreground glyph strokes in the ground-truth mask. Otherwise, it is considered a negative example.

We train a Radial Basis Function kernel SVM classifier using both foreground and background super-pixels in all training images. At test time, the trained classifier is applied to each region of a test image. The scores are converted into probabilities using a sigmoid function. Each super-pixel in an unknown image is therefore labeled with a probability to belong to a foreground glyph stroke.

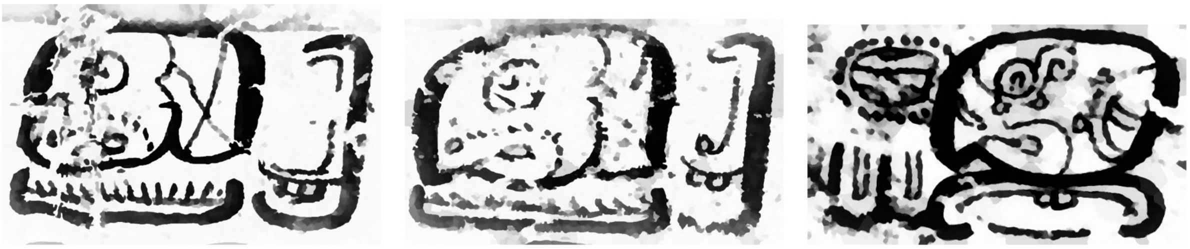


Fig. 4. Aggregated pixelwise probability maps of the first three example blocks in Figure 1. Darker color corresponds to higher foreground probability.

4.2.2 *Multiple Resolution Super-Pixels.*

The number of super-pixels extracted is a key factor that will affect the glyph binarization result. A larger number leads to binarization results better preserving details such as thin glyph strokes. However, it also keeps more background noise. In contrast, a smaller number leads to a coarser image presentation resolution and results in a cleaner binarization result. However, it will lose more delicate glyph stroke details. Therefore, the choice of the number of super-pixels is a compromise between glyph detail and background cleanness.

Given the nature of our glyph data, a large range of stroke widths is observed not only across the dataset but also within each single image (a glyph block in our case). Therefore, we propose to build our system using multiple resolution super-pixel regions. Precisely, a probability map from each different super-pixel resolution scale is first computed separately. Then the pixelwise probability maps from multiple resolution scales are aggregated to produce the final map:

$$\bar{p}(i) = \frac{1}{R}\sum_{r=1}^{R} p^r(i), \tag{2}$$

where $p^r(i)$ is the probability of pixel i being labeled as foreground in the rth resolution scale and $\bar{p}(i)$ is the aggregated probability score of pixel i. Figure 4 shows examples of aggregated probability maps. The darker the pixel, the higher the probability that it belongs to a glyph stroke. We can see that the aggregated probability map represents a compromise between delicate stroke details and background smoothness.

4.3 Fully Connected CRF

Label consistency between pixels is often obtained using neighboring pixels or regions as pairwise potentials in CRF models. Basic CRF models are usually composed of unary potentials on individual image components (pixels, patches, or regions) and pairwise potentials on neighboring components, to incorporate the labeling smoothness.

Such CRF models are limited in their ability to model long-range connections within an image. Hierarchical connectivity and higher-order potentials can be used in CRF models to improve the labeling consistency in a larger spatial range, as shown in Ladicky et al. [2009] and Kohli et al. [2009]. Recently, fully connected dense CRFs that model the pairwise dependencies between all pairs of pixels have been successfully used for image segmentation tasks [Krähenbühl and Koltun 2012].

In this article, we apply the model presented in Krähenbühl and Koltun [2012]. According to this model, the following energy function $E(x)$ is optimized:

$$E(x) = \sum_{i} \psi_u(x_i) + \sum_{i<j} \sigma_{x_i,x_j} \psi_p(x_i, x_j), \tag{3}$$

where the unary potential $\psi_u(x_i) = -\log \bar{p}(x_i|i)$ encodes the probability for pixel i to belong to class x_i, according to the recognition part of our model (Equation (2) in Section 4.2). While in the case of multi-label segmentation tasks, σ can be a carefully designed function that defines the compatibility between labels [Krähenbühl and Koltun 2012], here we use a simple binary value, that is, $\sigma_{x_i,x_j} = 0$ if $x_i = x_j$ (the labels assigned to i and j are the same) and 1 otherwise. The pairwise potential is defined as two Gaussian kernels,

$$\psi_p(x_i, x_j) = \omega_1 \exp\left(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}\right) + \omega_2 \exp\left(\frac{|p_i - p_j|^2}{2\theta_\gamma^2}\right), \tag{4}$$

with p_i and I_i being the position and RGB (red, green, and blue channel) values of pixel i, respectively. The first term encourages nearby pixels with similar color to share the same label. Note that due to this kernel, the smoothing of pixel labeling is non-isotropic and implicitly achieves some form of directional filtering, as the cost for having different labels at a given pixel is a function of the distribution of pixel values around that pixel. The degrees of nearness and similarity are controlled by parameters θ_α and θ_β. The second term aims to smooth the result by removing small isolated regions (the bandwidth θ_γ in this term is usually smaller than θ_α). The ω and θ parameters are optimized on the validation set.

5. EXPERIMENTS

In order to evaluate our proposed system, we first measure the pixelwise labeling results on a dataset for which the manually segmented clean raster images are available. We then conduct glyph retrieval on a subset of images for which the individual glyphs within each block have been manually segmented and labeled. We compare the retrieval results achieved using our automatically segmented binary images to those achieved using the manually segmented clean raster images. The goal is to understand and assess the effect of segmentation quality on a task relevant for epigraphers, namely glyph recognition via retrieval.

5.1 Datasets

We use two image datasets from ancient Maya codices, which were manually generated by epigraphers in our research team. See Section 3.2 for the data preparation conventions that we followed.

Maya Glyph Block Binarization (MGBB) Dataset. This dataset contains both the cropped raw images (see Figure 1(a)) and the manually generated clean raster forms (see Figure 1(b)) of 414 glyph blocks, among which 150 blocks are from the Dresden codex, 174 are from the Madrid codex, and the rest (90) are from the Paris codex. The sizes of our images are approximately 600 × 900. The percentage of these images with quality rankings from 0 to 4 are 3%, 3%, 5%, 25%, and 64%, respectively.

Maya Codex (MC) Dataset. It contains 759 individual glyphs segmented from 296 blocks, together with their annotated labels according to the Thompson catalog [Thompson 1962]. It is an extended dataset to the “codex dataset,” which we used in Hu et al. [2015]. We use this dataset for our retrieval experiments.

5.2 Glyph Stroke Extraction Results

The segmentation accuracy is evaluated on the MGBB dataset. The cropped raw images are used as input to our system. The clean raster images are used as ground truth to train our system in the case of training images or to evaluate the segmentation results in the case of test images.

We first study the effect of the number of super-pixels on the segmentation accuracy. We then compare the proposed system with methods popularly used in unsupervised low-level image segmentation

Table I. Segmentation Results with Various Super-Pixel Resolutions, Evaluated Using F-Measure (%)

Number of super-pixels	100	500	1000	1500	2000	2500	3000	merge
Appearance-only	78.96	92.17	92.88	92.91	92.87	92.81	92.76	93.50
Appearance + CRF	88.72	93.36	93.47	93.47	93.42	93.37	93.34	93.76

(*k*-means and Gaussian Mixture Model), document image binarization (Otsu's algorithm), and a state-of-the-art saliency detection framework [Perazzi et al. 2012]. The pixelwise labeling result of each image is evaluated with F-measure, which has been popularly used to evaluate the performance of document image binarization systems [Pratikakis et al. 2013],

$$F = \frac{2 \times Recall \times Precision}{Recall + Precision}, \quad (5)$$

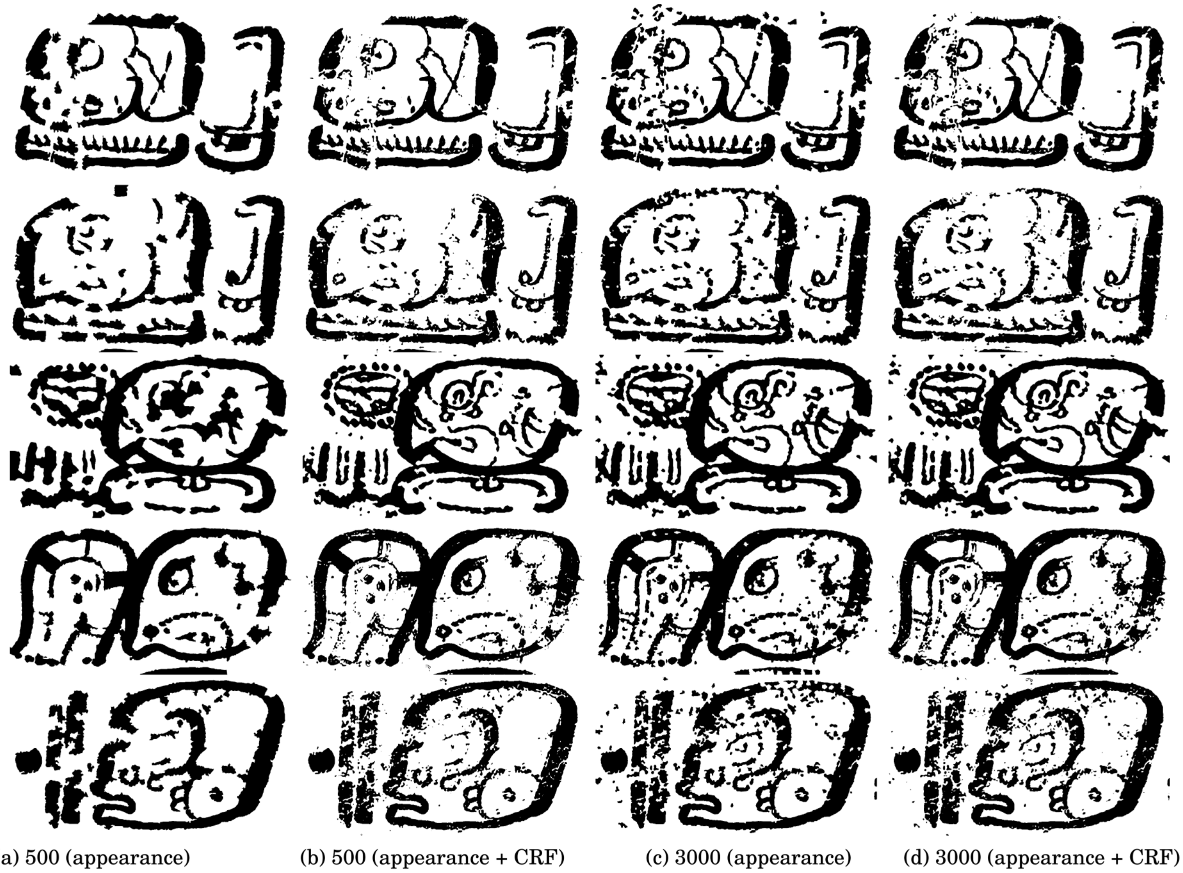
where $Precision = \frac{TP}{TP+FP}$ refers to the percentage of correctly assigned foreground (glyph stroke) pixels in the estimated result, and $Recall = \frac{TP}{TP+FN}$ corresponds to the fraction of correctly detected foreground pixels in relation to the number of foreground pixels in the ground-truth image. *TP*, *FP*, and *NP* denote the True Positive, False Positive, and False Negative values, respectively.

5.2.1 *Implementation Details.* We randomly split the MGBB dataset into two halves. One is used as the training set, and the other is used as the test set. Fivefold cross-validation is conducted on the training set to select the optimal parameters for the SVM classifier. The trained model is then applied on the test set to generate the probability map. We swap the training set with the test set and repeat the same process to compute the probability map for all images. The CRF model is then conducted to improve the labeling consistency. In our experiments, we use the following parameters in the CRF model (see Equation (4)): $\omega_1 = 2$, $\omega_2 = 1$, $\theta_\alpha = 40$, $\theta_\beta = 10$, $\theta_\gamma = 3$. We repeat this process 10 times. The average F-measure score achieved on all images is used to evaluate the segmentation accuracy of our proposed framework.

We conduct experiments with various image partitioning details, that is, the number of super-pixels (100; 500; 1,000; 1,500; 2,000; 2,500; 3000), to study their impact on the results. We then test the multiple-resolution-scale-based result by aggregating all seven aforementioned super-pixel scales. A dense CRF model is then applied on each setting.

From the results in Table I, we can see that when only the appearance model is applied (first row), the segmentation accuracy improves with the increasing number of super-pixels extracted, from 100 to 1,500, and then the performance starts to drop slightly (saturate). This is likely due to the fact that a small amount of super-pixels is not enough to represent glyph stroke details, while a large amount may preserve unwanted background noise. See Figures 5(a) and (c) for visual examples. When the CRF model is applied (second row in Table I), we observe that this sophisticated post-processing brings more notable performance improvement to the lower super-pixel resolution-based results (e.g., 100, 500) than to the higher-resolution ones. Similarly, by comparing Figure 5(b) to Figure 5(a), we can observe a noticeable result improvement, while only slightly improved stroke details and background smoothness can be observed by comparing Figure 5(d) to Figure 5(c).

From the last column of Table I, we can see that, by simply aggregating the probability maps over different super-pixel resolution scales, our method achieves a higher accuracy than any single resolution-based method. This result is further improved by incorporating the CRF model (although by a small margin). Comparing the results in Figure 6(b) to Figure 6(a), we can see that slightly more delicate details are preserved (although this could also bring more holes and broken lines to glyph strokes from the degraded original images). Additionally, our final results (Figure 6(b)) show overall better visual



(a) 500 (appearance) (b) 500 (appearance + CRF) (c) 3000 (appearance) (d) 3000 (appearance + CRF)

Fig. 5. Visual comparison of the segmentation results using different numbers of super-pixels (500 or 3,000) and segmentation models (appearance-based method alone or when the CRF model is considered).

quality in terms of delicate stroke detail preservation and background smoothness than any single scale result in Figure 5.

Figure 7 shows two examples, whose segmentation results are among the lowest of the whole query set (82.9% in both cases with F-measure). The two raw images show low contrast between foreground and background pixels at local detailed areas (especially the inner part of the glyphs), which makes it challenging for automatic segmentation algorithms.

5.2.2 *Comparison to Baseline Methods.* Pixel clustering is one of the most simple image segmentation methods. Here we apply k-means and Gaussian Mixture Model (GMM) for our glyph stroke extraction task. Pixels of each individual image are clustered into two categories in RGB feature space.

As a strong baseline, Otsu's algorithm [Otsu 1979] has been popularly used for document image binarization. Compared to local thresholding methods, Otsu's method does not require a pre-defined local window size, which in our case is difficult to set given the wide variety of stroke widths and background noise. We first convert each color image into grayscale. Otsu's algorithm is then used to estimate a global threshold to transform the grayscale image into a binarized form.

Additionally, we applied the adaptive thresholding [Sauvola and Pietikäinen 2000] algorithm, which achieves promising results in the document binarization evaluation article of Stathis et al. [2008]. A

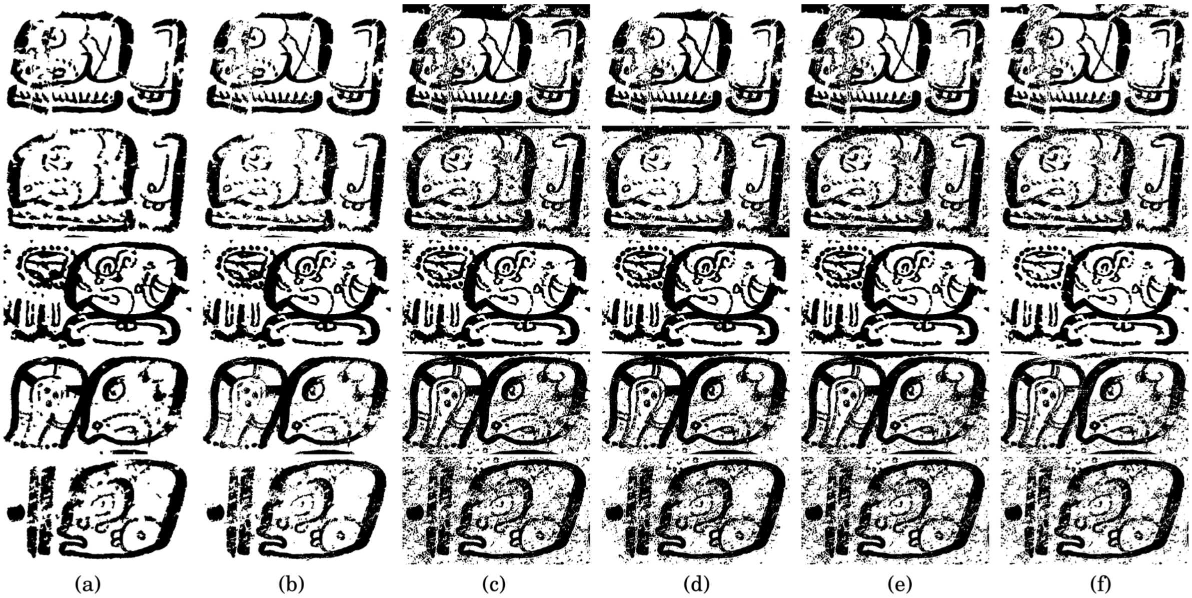


Fig. 6. Visual comparison of the segmentation results using different methods: (a) The proposed method with only the appearance component; (b) the proposed method with the CRF model; (c) *k*-means; (d) saliency detection; (e) Otsu's algorithm [Otsu 1979]; (f) adaptive thresholding algorithm [Sauvola and Pietikäinen 2000].

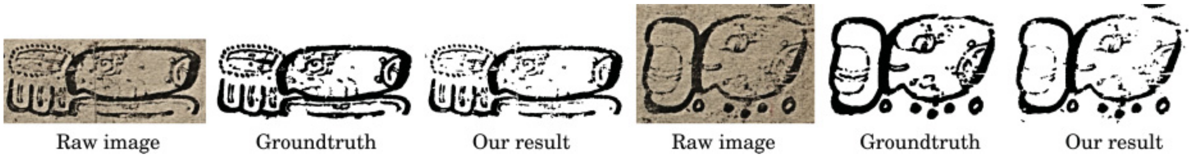


Fig. 7. Two examples of low segmentation results, based on F measure, using our method. Raw images were manually cropped and enhanced from the Dresden codex [Förstemann 1880] by Carlos pallán Gayol (supported by the German Research Foundation through the MAAYA project, University of Bonn, used with permission).

threshold is estimated at each pixel based on mean and standard deviation within a local window around the pixel. The binarization result is strongly affected by the local window size. In our experiment, we picked the local window size that gives the highest binarization result (average F-measure) on our database. The selected local window size is 7 times smaller than the size of the input image.

Finally, extracting glyph strokes from noisy background can also be considered as a saliency detection problem. Foreground glyph strokes are considered as salient objects of the raw glyph block image. A region-based saliency filter technique [Perazzi et al. 2012], which achieves state-of-the-art saliency detection accuracy, has been used on our data as a binary segmentation method.

Table II shows the segmentation results using the aforementioned algorithms individually to compare with the result achieved using our method. From the results, we can see that our method performs the best, followed by the local adaptive algorithm. Otsu's algorithm achieves a similar accuracy with *k*-means and GMM. The saliency detection-based method achieves the lowest accuracy among all, which could be due to the fact that there are several cases where the glyph strokes are detected as background while the actual image background is considered as a salient object.

Table II. Segmentation Results Using Different Algorithms

Methods	k-means	GMM	Global thresholding [Otsu 1979]	Local adaptive thresholding [Sauvola and Pietikäinen 2000]	Saliency [Perazzi et al. 2012]	Our method
F-measure (%)	89.74	89.20	89.95	91.29	84.98	93.76

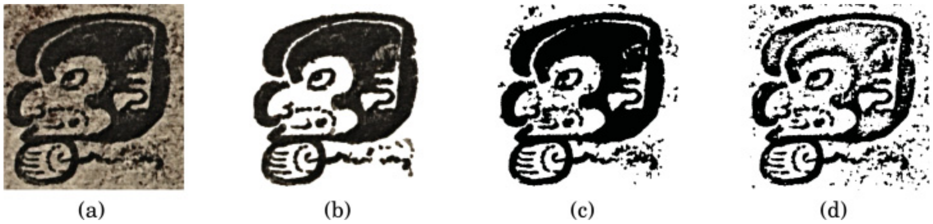


Fig. 8. (a) raw image (manually cropped by Carlos pallán Gayol from the Dresden Codex, supported by the German Research Foundation through the MAAYA project, University of Bonn, used with permission); (b) manually segmented groundtruth image; (c) our method; (d) adaptive thresholding algorithm [Sauvola and Pietikäinen 2000].

Figure 6 shows a selection of results for visual comparison of the segmentation output. We can see that, compared to other algorithms, our proposed method can remove most of the background noise, as well as extra noisy strokes close to the boundary, and at the same time preserves delicate details. To further illustrate the effectiveness of our method, Figure 8 shows a typical example where the adaptive thresholding algorithm failed due to its various stroke widths, the trained window size being not large enough to cover enough background and foreground areas. Our method is able to produce reasonable result.

5.3 Glyph Retrieval Results

Maya hieroglyphic analysis requires epigraphers to spend a significant amount of time browsing existing catalogs to identify individual unknown glyphs from a block. Our previous work proposed an automatic Maya hieroglyph retrieval algorithm [Hu et al. 2015] to match unknown query glyphs to a database of labeled catalog sign examples to facilitate the daily work of epigraphers.

Document image binarization is a necessary pre-processing step for most document analysis tasks, such as glyph retrieval. In order to evaluate our presented binarization method in the light of a practical application, we conduct glyph retrieval using binary images produced either manually (Figure 1(d)) or by several different automatic approaches, that is, our presented method (Figure 6(b)), Otsu's global thresholding method (Figure 6(e)), and the adaptive thresholding algorithm (Figure 6(f)). The motivation of this experiment is twofold: First, evaluate how much degradation of the retrieval performance results from using our presented automatic binarization method compared to using the manual approach, and, second, study how the retrieval result is affected by using different automatic binarization methods and how much a difference in binarization performance translates into a difference in retrieval performance.

5.3.1 *Retrieval System.* We use the MC dataset. Individual glyphs within each block are considered as queries to match with a database of known glyphs (i.e., the retrieval database) following the system proposed in Hu et al. [2015]. The top-ranked results are returned to the user as recommended labels to the query.

Segmenting individual glyphs from a block is in itself a challenging task and not the focus of this article. We apply a tight bounding box over each individual glyph that epigraphers manually segmented

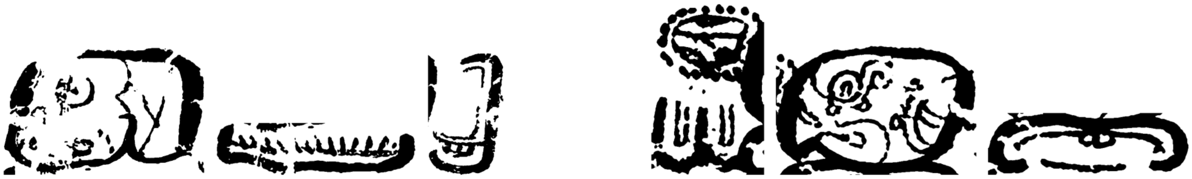


Fig. 9. Individual glyphs cropped from the generated binary images of the same raw black sources as the ones shown in Figure 2.

from clean raster images (see Figure 2) and use it on the generated binary images, as shown in Figure 9. We can see that this cropping method introduces extra strokes from the neighboring glyphs in a block, which can further affect the retrieval results.

The retrieval database is composed of 1,487 glyph images belonging to 892 different sign categories, scanned and cropped from the Thompson catalog [Thompson 1962]. Each category is usually represented by a single example image. Sometimes multiple examples are included; each illustrates a different visual instance or a rotation variant of the represented sign.

Following Hu et al. [2015], our retrieval pipeline includes two main components: appearance-based (shape-based) glyph similarity matching, and a statistical-language-model-based re-ranking.

5.3.1.1 *Appearance-Based Glyph Retrieval*. Given a query glyph, we first pre-process it into thin lines using mathematical morphological operations. Histogram-of-Orientation Shape Context features [Roman-Rangel et al. 2011] are then extracted on evenly distributed pivot points along the lines. We then follow the Bag-of-Visual-Words (BoVW) representation, where descriptors are quantized as visual words based on the vocabulary obtained through k-means clustering on the set of descriptors extracted from the retrieval database. A histogram representing the count of each visual word is then computed as a global descriptor for each glyph. Let S_i denote the glyph label we want to assign to a query glyph G_i, and then the shape similarity score $Score^{shape}(S_i = D)$ is computed using the L_1 norm distance between the BoVW-based global shape representation of G_i and sign D in the retrieval database.

5.3.1.2 *Incorporating Glyph Co-Occurrence Information*. Maya glyph blocks were frequently composed of combinations of individual signs. Glyph co-occurrence within single blocks therefore encode valuable context information, which we aim at exploiting. To do so, our methodology converts each block into a linear string of individual glyphs, according to the reading order (see Figure 2 and Figure 9). Thus, denoting by $G_{1:n} = [G_1, \ldots, G_i, \ldots, G_n]$ the string of a block, and by $S_{1:n}$ the corresponding sequence of labels, we would like to assign to $G_{1:n}$, and the score of the glyph G_i being recognized as D can then be defined to take into account all observed glyphs in the string, according to

$$Score^{shape+context}(S_i = D) = \max_{S_{1:i-1;i+1:n}} P(S_{1:n}|G_{1:n}), \tag{6}$$

which can be interpreted as follows: The score of a candidate sign D to be the label of G_i will not only depend on whether D is visually similar to G_i but also on whether D normally co-occurs with signs that are visually similar to the other glyphs (G_j, $j \neq i$) in the block. To compute the above score, we assume the glyph string to follow a first-order Markov chain. The co-occurrence information is then coded as a language model (referred to as "LM") represented by the transition probability $P(S_j|S_{j-1})$ learned from labeled data (see Vail model in [Hu et al. 2015]). Under this assumption, the score in Equation (6) can be efficiently computed using the Viterbi algorithm.

5.3.2 *Experimental Set-Ups.* According to the previous sections, several experimental set-ups are tested and evaluated. First, binary images produced by various methods are considered: manually

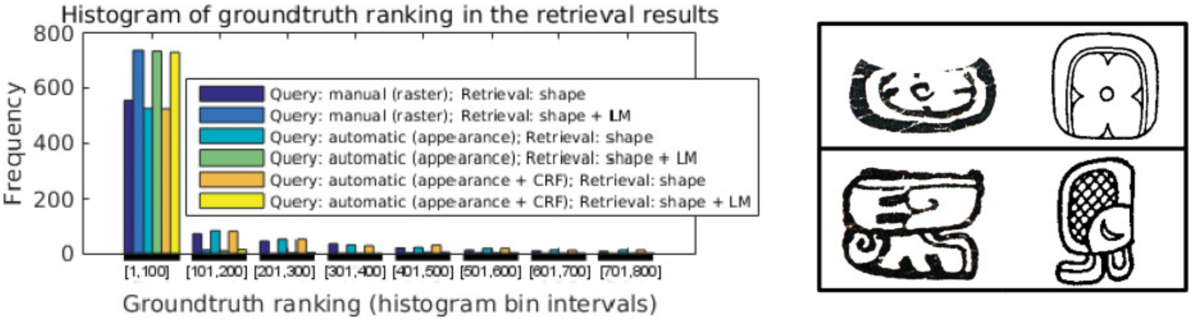


Fig. 10. Left: Histogram of the ground-truth ranking in the retrieval results, that is, the number of queries, whose rank of the correct match in the retrieval database are returned within the [1,100], [101,200], . . . , [701,800] intervals. Right: Two problematic glyph queries (first column) and their correct matched ground-truth glyphs in the retrieval database (second column) manually annotated by epigraphers in our team.

generated clean raster images (referred to as “raster”), binary images automatically generated using our proposed method with only the appearance feature (“appearance”) and when the CRF model is applied (“appearance + CRF”), binary images produced using Otsu’s [Otsu 1979] global thresholding algorithm (“Otsu”), and an adaptive thresholding method [Sauvola and Pietikäinen 2000] (“adaptive”). Second, for each query type, we considered the two retrieval systems described in Section 5.3.1, that is, using only the shape feature (referred to as “shape”), and when the glyph co-occurrence information is incorporated (“shape + LM”). Note that although the co-occurrence information is only available when the neighboring glyphs of a query is available, we present these results, as our objective is to evaluate our binarization method in different retrieval scenarios, that is, shape alone and when context information is considered.

5.3.3 *Results and Discussion.* The proposed retrieval task is very challenging for several reasons. First, different signs often share similar visual features. Second, glyphs of the same sign category vary with time, location, and individual styles. Finally, the surviving Maya scripts have often lost their visual quality over time.

Due to the experimental set-up of our retrieval database (catalog sign examples), most queries have only one correct match in the retrieval database. The higher the ground truth is ranked in the returned results, the better the retrieval accuracy. Figure 10 (left) shows the histogram of the ground-truth rankings.

From the histogram, we can see the following. First, our automatically generated binary images achieves slightly lower results to that of the manually produced raster images. Second, the correct match for most queries are returned in the top 100 results. There are only a few cases for which the correct match is ranked beyond 500. These few poor results are usually caused by problematic queries, such as degraded glyph strokes with low visual quality or large visual differences between the query glyphs and their ground-truth signs in the catalog (see Figure 10 (right) for examples). Visual differences are mainly caused by two reasons: (1) the Maya civilization spanned a long period of time, during which the writing of each glyph could have varied with time and location and (2) ancient Maya scribes followed flexible artistic conventions to render each sign, which may have resulted in visually different glyphs.

Figure 11 shows the percentage of queries with ground-truth ranking in the top 100 returned results. First, for all query types, retrieval results are significantly improved by incorporating the glyph co-occurrence information (“LM”). Second, according to this measure, when only the shape feature

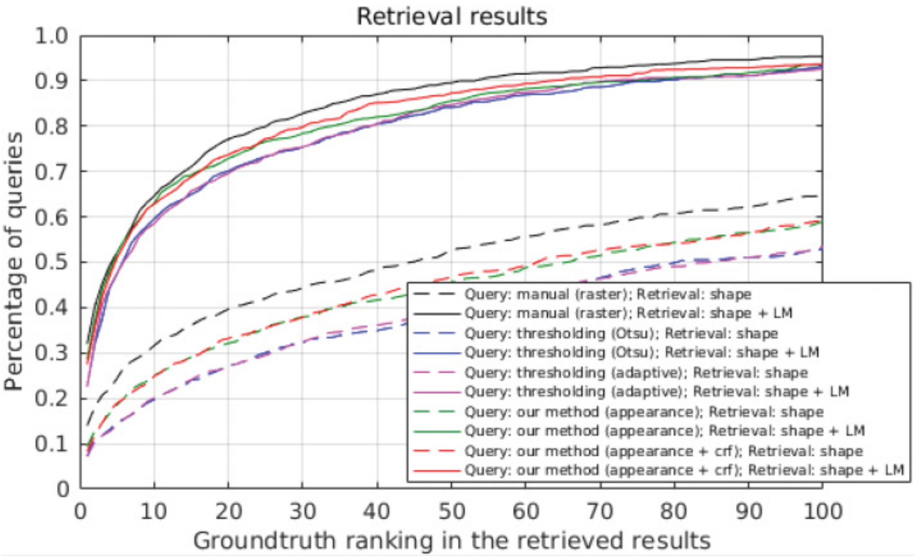


Fig. 11. Curves of the percentage of queries with ground-truth ranking in the top 100 returned results, that is, the percentage of queries whose correct match in the retrieval database are returned within the first 1, 2, . . . , 100 returned results.

Table III. Retrieval Results, That Is, Average (and Median) Values of the Ground-Truth Rankings in the Returned Results over All the Query Images in the MGBB Dataset

		Thresholding		Our method	
Query	Manual	Otsu	Local adaptive	appearance-only	appearance + CRF
Shape-only	120.6 (45)	169.0 (85)	168.1 (81)	142.1 (64)	141.4 (62)
Shape + LM	21.3 (5)	30.9 (6)	30.6 (6)	28.1 (5)	27.2 (5)

is used, compared to results achieved using the manually segmented binary images, results using binary images automatically generated by our method are worse by around 20% (at rank 10), and by 40% when using the thresholding methods. However, the difference becomes less significant when the glyph co-occurrence information (“LM”) is considered. Third, when the glyph co-occurrence information is considered, around 30% of queries achieve their correct match in the first returned results, and around 63% of queries achieve their correct match in the top 10 returned results using our binarization method. In contrast, around 23% and 59% of queries achieve their correct match in the first and top 10 retrieved results when thresholding algorithms are used. Fourth, similar retrieval results are achieved using Otsu’s global thresholding and the adaptive thresholding algorithms.

Finally, Table III displays the average and median values of the ground-truth rankings over the whole query set (median values are shown in brackets). The smaller the average/median ranking, the better the retrieval results. We can see that our automatically segmented binary images produce comparable retrieval results to those achieved by manually segmented clean raster images when the statistic language model is incorporated. Considering the CRF model in the segmentation system slightly improves the retrieval accuracy. However, after applying the language model, this difference becomes negligible. Retrieval results achieved using binary images generated by our method are higher than those achieved using the thresholding algorithms. It is worth noticing that the average ranking results are blasted by the problematic queries, as explained in previous paragraphs. The median ranking is a more robust measurement in our case. We can see that by incorporating glyph co-occurrence

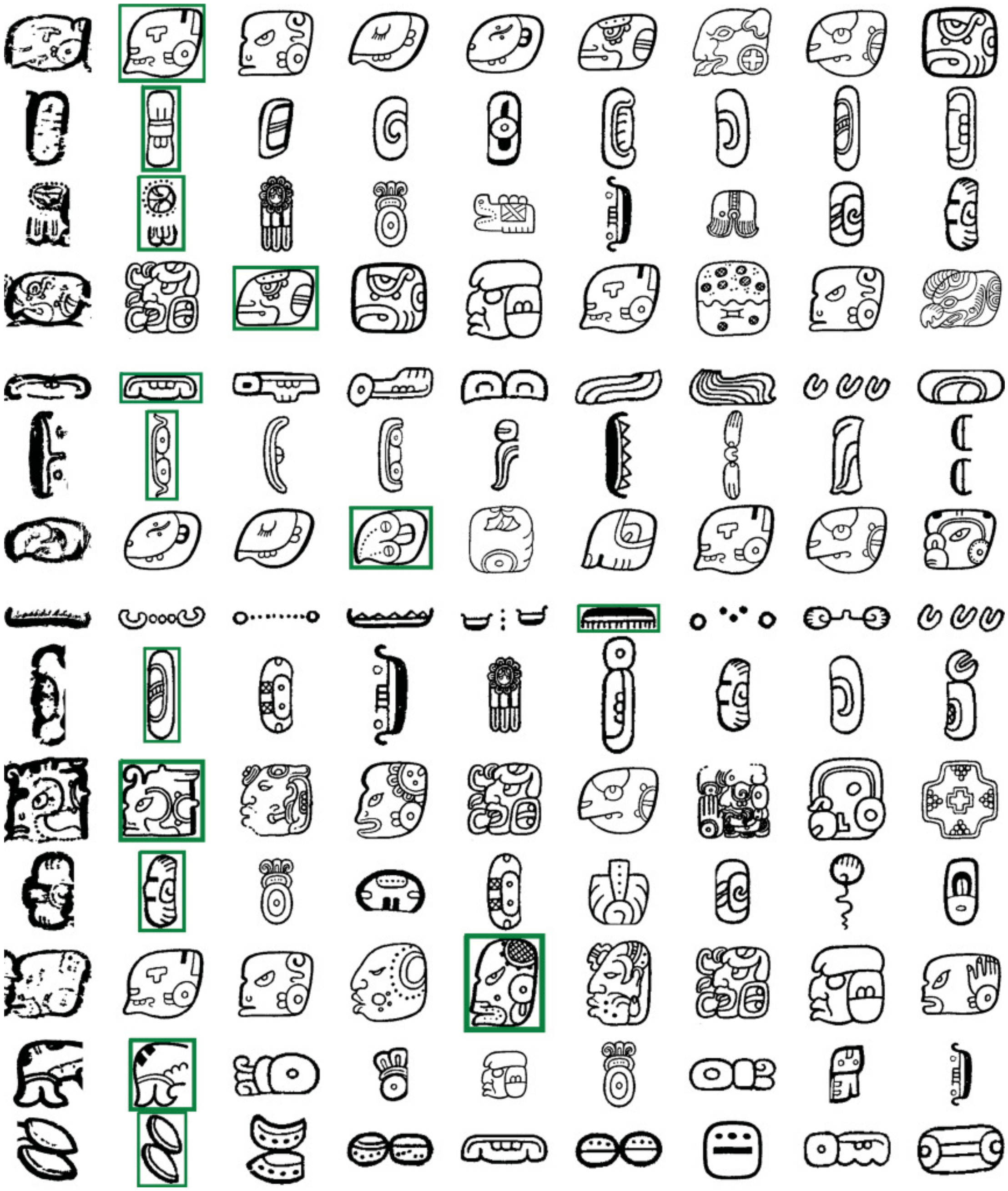


Fig. 12. Examples of retrieval results. The first column shows query glyphs, automatically extracted from raw images (raster images compiled by Carlos Pallán Gayol, MAAYA project, University of Bonn, from the Dresden Codex [Förstemann 1880]) using our proposed method. Their top returned results, of the 1,487 glyph examples in the retrieval database (images from the Thompson hieroglyph catalog [Thompson 1962]), are shown from left to right in each row. The correct match are highlighted in green bounding boxes.

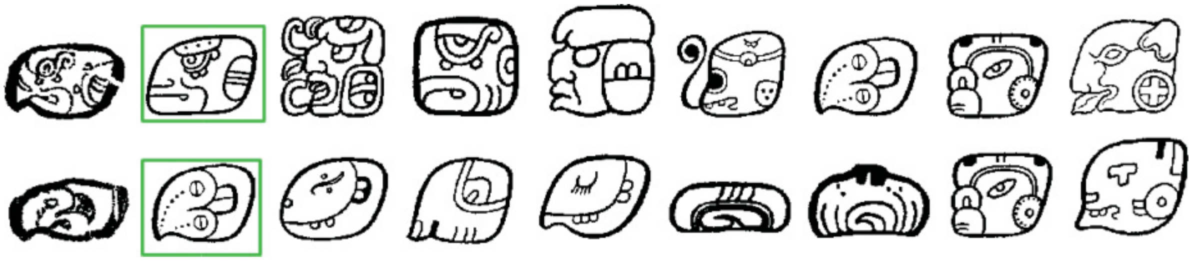


Fig. 13. Example retrieval results using manually segmented binary images as queries (manually cropped and produced by Carlos Pallán Gayol, funded by the German Research Foundation through the MAAYA project, University of Bonn, used with permission). The first column shows query glyphs. Note that the two queries are manually segmented from the same images as the fourth and the seventh queries in Figure 12. Their top returned results are shown from left to right in each row. The correct match are highlighted in green bounding boxes.

information, our automatically produced binary images generate the same median ranking results to those achieved by the clean raster images manually produced by epigraphers.

Figure 12 shows example queries (first column) automatically generated using the "appearance + CRF" method and their top returned retrieval results using "shape + LM" method (second-to-last columns in each row). These examples illustrate that our system is able to effectively search the correct match in the top returned results from a database of catalog signs, where visually similar examples often exist. To visually compare retrieval results obtained from our automatically generated binary images to those obtained using the manually segmented ones, we show two manually generated query examples and their top returned results in Figure 13. Note that the first example in Figure 13 is the counterpart of the fourth example in Figure 12. They are the same query example segmented using different methods, that is, manually or by our presented automatic binarization method. Similarly, the second example in Figure 13 is the counterpart of the seventh example in Figure 12.

6. CONCLUSION

We introduce an automatic Maya hieroglyph stroke extraction system through region-based image segmentation. Our method can, in principle, be used in any image binarization task, especially in cases where thresholding algorithms do not work well, such as images of degraded ancient documents with various stroke widths and local details.

Our system consists of an appearance-based classification model and a fully connected CRF framework to improve the labeling consistency. Multiple resolution super-pixel regions are used to cope with various stroke widths both across the dataset and within individual images, as well as to remove various types of background noise. Experimental results show that our method preserves delicate historic Maya glyph stroke details and reduces background noise.

Our model is learned on a relatively small and manually cleaned glyph collection. The learned model can then be used on raw data to automatically extract valuable historic glyph strokes from noisy background. This might further facilitate the automatic analysis of ancient Maya documents as more data become available and could potentially help the decipherment of ancient Maya hieroglyphs.

Comparable glyph retrieval results are achieved between our automatically produced binary images and manually segmented data. This result is particularly encouraging, because it justifies the use of our system on the full codex data, which will significantly save the time of epigraphers.

Finally, two image datasets of Maya hieroglyphs were used. They include a subset of glyph blocks cropped from the three surviving ancient Maya codices as well as their manually produced clean raster images and a smaller set of individual glyphs segmented from each block.

Future work could include a comparative analysis of the performance of our method on other data sources, such as Maya monument data and ancient Chinese calligraphy documents. Other future works include automatic Maya document layout analysis and segmenting individual glyphs from blocks to enable more advanced functionalities for automatic Maya document analysis and understanding.

ACKNOWLEDGMENTS

We thank the Swiss National Science Foundation (SNSF) for support through the MAAYA project. We also thank Carlos Pallán Gayol and Guido Krempel (University of Bonn), and Jakub Spotak (Comenius University in Bratislava) for producing the data used in this paper (supported by the German Research Foundation DFG), and for sharing it for academic research.

REFERENCES

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurélien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2012. SLIC Superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 11 (2012), 2274–2282.

Arwa Mahmoud AL-Khatatneh, Sakinah Ali Pitchay, and Musak Kasim Al-qudah. 2015. Compound binarzation for degraded document images. *ANPR Journal of Engineering and Applied Sciences* 10, 2 (2015), 594–599.

Pablo Arbelaez, Michael Maire, Charless C. Fowlkes, and Jitendra Malik. 2011. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 5 (2011), 898–916.

John Bernsen. 1986. Dynamic thresholding of gray level image. In *ICPR*. 1251–1255.

Yuri Boykov and Marie-Pierre Jolly. 2001. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*. 105–112.

Dorin Comaniciu and Peter Meer. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 24, 5 (2002), 603–619.

Antonio Criminisi, Toby Sharp, Carsten Rother, and Patrick Pérez. 2010. Geodesic image and video editing. *ACM Trans. Graph.* 29, 5 (2010), 134.

Gabriela Csurka and Florent Perronnin. 2011. An efficient approach to semantic segmentation. *Int. J. Comput. Vis.* 95, 2 (2011), 198–212.

E. B. Evrenov, Y. Kosarev, and B. A. Ustinov. 1961. *The Application of Electronic Computers in Research of the Ancient Maya Writing*. USSR, Novosibirsk.

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. 2004. Efficient graph-based image segmentation. *Int. J. Comput. Vis.* 59, 2 (2004), 167–181.

Andreas Fischer, Micheal Baechler, Angelika Garz, Marcus Liwicki, and Rolf Ingold. 2014. A combined system for text line extraction and handwriting recognition in historical documents. In *International Workshop on Document Analysis Systems*. 71–75.

Ernst Wilhelm Förstemann. 1880. Die Maya-Handschrift der königlichen Bibliothek zu Dresden. Leipzig : Verlag der Naumann'schen Lichtdruckerei.

Morris Franken and Jan C. van Gemert. 2013. Automatic egyptian hieroglyph recognition by retrieving images as texts. In *ACM MM*. 765–768.

Basilios Gatos, Konstantinos Ntirogiannis, and Ioannis Pratikakis. 2009. ICDAR 2009 document image binarization contest (DIBCO 2009). In *International Conference on Document Analysis and Recognition*. 1375–1382.

Basilios Gatos, Ioannis Pratikakis, and Stavros J. Perantonis. 2008. Improved document image binarization by using a combination of multiple binarization techniques and adapted edge information. In *ICPR*. 1–4.

Michael Gleicher. 1995. Image snapping. In *Annual Conference on Computer Graphics and Interactive Techniques*. 183–190.

Rui Hu, Gulcan Can, Carlos Pallan Gayol, Guido Krempel, Jakub Spotak, Gabrielle Vail, Stéphane Marchand-Maillet, Jean-Marc Odobez, and Daniel Gatica-Perez. 2015. Multimedia analysis and access of ancient maya epigraphy: Tools to support scholars on maya hieroglyphics. *IEEE Sign. Process. Mag.* (2015), 75–84.

Rui Hu, Diane Larlus, and Gabriela Csurka. 2012. On the use of regions for semantic image segmentation. In *ICVGIP*. 51.

Pushmeet Kohli, Lubor Ladicky, and Philip H. S. Torr. 2009. Robust higher order potentials for enforcing label consistency. *Int. J. Comput. Vis.* 82, 3 (2009), 302–324.

Philipp Krähenbühl and Vladlen Koltun. 2012. Efficient inference in fully connected CRFs with gaussian edge potentials. *CoRR* abs/1210.5644 (2012).

Lubor Ladicky, Christopher Russell, Pushmeet Kohli, and Philip H. S. Torr. 2009. Associative hierarchical CRFs for object class image segmentation. In *International Conference on Computer Vision*. 739–746.

Shijian Lu, Bolan Su, and Chew Lim Tan. 2010. Document image binarization using background estimation and stroke edges. *IJDAR* (2010), 303–314.

Xiaoqing Lu, Zhi Tang, Yan Liu, Liangcai Gao, Ting Wang, and Zhipeng Wang. 2013. Stroke-based character segmentation of low-quality images on ancient chinese tablet. In *International Conference on Document Analysis and Recognition*. 240–244.

Martha J. Macri and Gabrielle Vail. 1901. *The New Catalog of Maya Hieroglyphs, Vol. II, The Codical Texts*. University of Oklahoma Press.

Nikolaos Mitianoudis and Nikolaos Papamarkos. 2015. Document image binarization using local features and gaussian mixture modeling. *Image Vis. Comput.* (2015), 33–51.

Reza Farrahi Moghaddam and Mohamed Cheriet. 2012. AdOtsu: An adaptive and parameterless generalization of Otsu's method for document image binarization. *Pattern Recogn.* (2012), 2419–2431.

Wayne Niblack. 1990. *An Introduction to Digital Image Processing*. 115–116 pages.

Nobuyuki Otsu. 1979. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybernet.* (1979), 62–66.

Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. 2012. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*. 733–740.

Ioannis Pratikakis, Basilis Gatos, and Konstantinos Ntirogiannis. 2013. ICDAR 2013 document image binarization contest (DIBCO 2013). In *International Conference on Document Analysis and Recognition*. 1471–1476.

Jos B. T. M. Roerdink and Arnold Meijster. 2000. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundam. Inform.* 41, 1–2 (2000), 187–228.

Edgar Roman-Rangel, Carlos Pallan-Gayol, Jean-Marc Odobez, and Daniel Gatica-Perez. 2011. Searching the past: An improved shape descriptor to retrieve Maya hieroglyphs. In *ACM MM*. 163–172.

Azriel Rosenfeld and Pilar de la Torre. 1983. Histogram concavity analysis as an aid in threshold selection. *IEEE Trans. Syst. Man Cybernet.* (1983), 231–235.

Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. 2004. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* 23, 3 (2004), 309–314.

Philippe Salembier, Albert Oliveras, and Luis Garrido. 1998. Antiextensive connected operators for image and sequence processing. *IEEE Trans. Image Process.* 7, 4 (1998), 555–570.

J. Sauvola and M. Pietikäinen. 2000. Adaptive document image binarization. *Pattern Recogn.* 33 (2000), 225–236.

Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 8 (2000), 888–905.

Jamie Shotton, Matthew Johnson, and Roberto Cipolla. 2008. Semantic texton forests for image categorization and segmentation. In *CVPR*.

Pyrrhos Stathis, Ergina Kavallieratou, and Nikos Papamarkos. 2008. An evaluation technique for binarization algorithms. *J. UCS* 14, 18 (2008), 3011–3030. DOI:http://dx.doi.org/10.3217/jucs-014-18-3011

Bolan Su, Shijian Lu, and Chew Lim Tan. 2011. Combination of document image binarization techniques. In *International Conference on Document Analysis and Recognition*. 22–26.

Bolan Su, Shijian Lu, and Chew Lim Tan. 2013. Robust document image binarization technique for degraded document images. *IEEE Trans. Image Process.* (2013), 1408–1417.

J. Eric S. Thompson. 1962. *A Catalog of Maya Hieroglyphs*. University of Oklahoma Press.

Gabrielle Vail and Christine Hernández. 2013. The Maya Hieroglyphic Codices. Retrieved from http://www.mayacodices.org/.

Jakob J. Verbeek and Bill Triggs. 2007. Scene segmentation with CRFs learned from partially labeled images. In *Annual Conference on Neural Information Processing Systems*. 1553–1560.

Luc Vincent and Pierre Soille. 1991. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Mach. Intell.* 13, 6 (1991), 583–598.

Xiangyang Wang, Xian-Jin Zhang, Hong-Ying Yang, and Juan Bu. 2012. A pixel-based color image segmentation using support vector machine and fuzzy *C*-Means. *Neur. Netw.* 33 (2012), 148–159.

Lin Yang, Peter Meer, and David J. Foran. 2007. Multiple class segmentation using A unified framework over mean-shift patches. In *CVPR*. 1–8.

G. Zimmermann. 1956. *Die Hieroglyphen Der Maya Handschriften. Abhandlungen Aus Dem Gebiet Der Auslandskunde*. Band 62- Reihe B, Universität Hamburg. Cram, De Gruyter & Co.

Received March 2016; revised June 2016; accepted September 2016