Indexing Mayan Hieroglyphs with Neural Codes

Edgar Roman-Rangel
CVMLab - University of Geneva. Switzerland.
edgar.romanrangel@unige.ch

Stephane Marchand-Maillet
CVMLab - University of Geneva. Switzerland.
stephane.marchand-maillet@unige.ch

***Abstract*—We present an approach for unsupervised computation of local shape descriptors, which relies on the use of linear autoencoders for characterizing local regions of complex shapes. The proposed approach responds to the need for a robust scheme to index binary images using local descriptors, which arises when only few examples of the complete images are available for training, thus making inaccurate the learning process of parameters of traditional neural networks schemes. Given the possibility of using linear operations during the encoding phase, the computation of the proposed local descriptor can be fast once the parameters of the encoding function are learned. After conducting a vast search, we identified the optimal dimensionality of the resulting local descriptor to be of only 128 dimensions, which allows for efficient further operations on them, such as the construction of bag representations with purposes of shape retrieval and classification. We evaluated the proposed approach indexing a collection of complex binary images, whose instances contain compounds of hieroglyphs from the ancient Maya civilization. Our retrieval experiments show that the proposed approach achieves competitive retrieval performance when compared with hand-crafted local descriptors.**

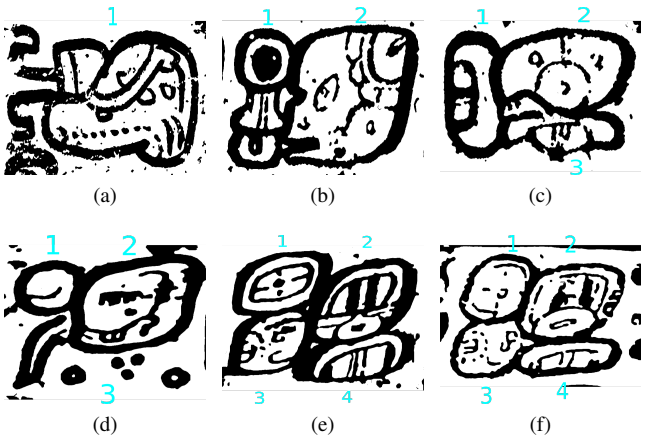

Fig. 1. Examples of Mayan glyph-blocks with probable reading order indicated by the cyan numbers. (a): 1 single glyph-sign; (b): 2 glyph-signs; (c) and (d): two different blocks with 3 glyph-signs; and (e) and (f): two instances of the same block with 4 glyph-signs.

I. Introduction

Neural Network (NNs) have regained much popularity in recent years for addressing many challenges in computer vision, including image classification, image retrieval, and image detection [1]. In particular, they have set the state-of-the-art on learning robust representations for automatic classification and retrieval of shapes [2], [3], [4].

Besides the growth in available computational power, another characteristic that makes NNs so successful is the large datasets that are often used for learning their parameters, i.e., given a large amount of training examples, NNs can learn a series of mapping functions able to represent highly complex visual structures. However, given the large number of parameter involved in a NN representation, their potential might easily become called into question for the cases when only relative small sets are available for training.

Our project seeks to develop computer vision tools to support the work of archaeologists and epigraphers, and more generally for Cultural Heritage preservation (i.e., Digital Humanities) [5]. Namely, we focus on the design of indexing methods for retrieval and visualization of visual instances of ancient inscriptions, and in particular, shapes representing hieroglyphs of the Mayan culture. More precisely, we deal with the problem of indexing glyph-blocks, this is, compounds of individual glyph-signs that are visually grouped to convey complex ideas (e.g., the notion of sentence in modern structured languages.). Fig. 1 shows examples of shapes representing glyph-blocks of the Maya writing system. As seen in Fig. 1, glyph-blocks are visually very complex shapes, as they can be composed by one or more individual glyph-signs, whose spatial arrangement might vary largely both in function of the scale of its components and their spatial location.

One particular problem that we face is that of having only a few couple of thousand images (i.e., only a few hundred instances per visual class), whose size is 500x700 pixels on average. This results in a very high dimensional input vector for traditional NN techniques (convolutional neural networks included [6]), and thus in a huge quantity of parameters to be learned with respect to the size of the dataset. One possible approach for using convolutional neural networks (CNN) is that of down-sampling the input image, at the cost of loosing visual details. Another possibility, consists of training the network on a larger and well balanced shape dataset, and then use the trained network on the dataset of interest. We will show in sec. IV that neither of these approaches are successful for our dataset.

To tackle these issues, we propose the use of linear autoencoders to generate "local descriptors" in an unsupervised manner, while using patches segmented from the input shape, such that non -or very little- resizing is required, and thus most of the visual details are preserved. Namely, no classification learning is required given the lack of class labels in the local shape visual structures. Instead, the autoencoder is trained to

reconstruct its own inputs, learning a compressed representation of them in the process. Our experiments show that the retrieval performance of our approach is: (1) higher when compared with Convolutional Neural Networks (CNN) which required down-sampling, and with Neural Networks trained on other datasets, i.e., the MNIST dataset [2]; and (2) comparable with the Histogram of Orientations Shape Context (HOOSC) [7], which is a hand-crafted robust descriptor developed for dealing with complex shapes. Also, the proposed approach allows to reconstruct the input image, which is not possible by using HOOSC, and that might be important for visualization purposes in epigraphic analysis.

Similar to our work, is the approach proposed by Wang et al., [4], where a set of stacked autoencoders is used to process Chinese characters. However, their dataset is large enough for effective learning of the parameters of the autoencoders. Another closely related work in analysis of Mayan hieroglyphs [3] focuses on learning local shape features via autoencoders by learning its parameters on a dataset of generic shapes, and then applying it to the Mayan glyphs. As we will show in sec. IV, this approach was not effective for our problem. Note that these two past works deal with images containing individual glyph-signs instead of compounds of them as we do.

The rest of this paper is organized as follows. Section II details the approach we followed to index complex shapes. Section III gives details of the data used in our experiments. Section IV details our experimental protocol and results. And section V presents our conclusions.

II. Approach

The proposed approach for shape descriptions consists of a three steps process: (1) local sampling, (2) local encoding, and (3) bag construction. Fig. 2 shows its pipeline.

A. Local sampling

Sampling the local patches that will be feed into the autoencoder requires: (1) localizing the center of the patch and its corresponding local spatial scope, i.e., the size of the patch; and (2) resizing them into a standard size that matches the input size of the autoencoder.

1) Point localization: To localize the center of the local patches, we follow the approach of dense sampling for local descriptors [8], and more specifically, the approach for uniform random sampling implemented by the well known Shape Context descriptor [9] and its related, more robust version, Histogram of Orientations Shape Context (HOOSC) [7]. Specifically, we estimate the medial axis of the shape of interest, from which we randomly sample a set of points. This sampling approach has proven effective [7] as long as points are uniformly distributed over the shape and the amount of points is large enough to compensate for their random position.

We also follow SC's approach for estimating an adequate size of the local patches, which will be the same for all local patches within a single input shape, but not necessarily across them. In practice, a patch will have a size $T \times T$ pixels, where T corresponds the average pairwise distance computed between all pairs of points randomly sampled from the shape. The resulting local patch will contain its point of interest (i.e., sampled point) centered within. Note that this size might be shorter for the case when the point of interest is located very close to any border of the image.

Fig. 2b shows 10 points randomly sampled from a glyph-block and their corresponding local spatial scopes. Note that these points are located towards the center of the traces of the shape, as they are extracted from its medial axis [7].

2) Resizing: The use of neural networks, including autoencoders, requires all inputs to have the same size. For the case of images, it is common to keep them within a few thousand pixels in order to have a relative small sets of model parameters, e.g., 28×28 or 50×50 pixels are common practices. However, glyph-blocks resized to such values result in very small images where most visual details are lost, which is one of the main reasons for using local patches.

More specifically, the expected size of a patch is 165.6 ± 27.3 pixels in our dataset. Therefore, it can be resized to a regular size while retaining most of its visual details. Namely, for our experiments, we resized all local patches to 50×50 pixels, thus 2500 input size for the autoencoder. Fig. 2c shows examples of cropped local patches rescaled to 50×50 pixels.

B. Local encoding

The key part of the proposed approach is encoding the local patch into the local descriptor. To obtain our local descriptor we used a linear autoencoder with one hidden layer, and full connectivity between input and hidden layer, as well as between hidden and output layer.

More specifically, the activation function $a(\cdot)$ for the j-th unit h_j in the hidden layer is,

$$a(h_j) = b_j^h + \sum_i \omega_{j,i}^h x_i, \tag{1}$$

where, b_j^h is the j-th component of a bias vector b^h, $\omega_{j,i}^h$ is the weight connecting the i-th unit in the input layer with the j-th unit in the hidden layer, and x_i corresponds to the i-th component in the input layer.

The number of hidden units h_j is one of the parameters of the network. Sec. IV shows the performance obtained with several combinations of such parameters. Note that the formulation in Eq. (1) allows for mapping back from the hidden layer to the input signal.

In turn, to compute the activation of the units o_k in the output layer of the autoencoder, we used the sigmoid function,

$$a(o_k) = \frac{1}{1 + e^{-z_k}}, \tag{2}$$

where,

$$z_k = b_k^o + \sum_j \omega_{k,j}^o a(h_j), \tag{3}$$

where, b^o is the bias vector for the output layer indexed with k, and $\omega_{k,j}^o$ is the weight connecting the j-th unit from the hidden layer to the k-th unit in the output layer.

(a) Input shape (b) Sampling (c) Resizing (d) Encoding

Fig. 2. Local description of Mayan glyph-blocks with neural codes. (a): Input shape; (b): red dots indicate the points randomly sampled from the medial axis of the shape, note that they are uniformly distributed over the shape, 10 of these points are randomly chosen and marked with blue crosses to illustrate their respective local spatial scopes, which are bounded by the blue circles; (c): resizing of the local patches to standard input size (50×50) for the autoencoder, these examples correspond to the 10 points in blue in (b); (d) the resulting local descriptors encoded into 128 dimensions, one row per example in (c).

We noticed empirically that such combination of activation functions achieves slightly lower reconstruction error as compared with the used of tight weights [10]. Fig. 2d shows examples of the encoded local descriptors.

Training: To train the autoencoder, we used a subsets of E patches randomly sampled from instances of the glyph-blocks. We used stochastic gradient descent with batches of $0.1E$ patches, and updated the parameters using back-propagation. The cost function that we optimized is,

$$J(\Omega, B) = \frac{1}{2} \sum_{e} \|x^{(e)} - o^{(e)}\|_2^2 + \lambda \sum_{i,j,k} \left(\Omega^{i,j,k}\right)^2, \qquad (4)$$

where the parameters Ω and B denote, respectively, the sets of weight parameters $\Omega = \{\omega^h, \omega^o\}$ and of bias terms $B = \{b^h, b^o\}$ for the hidden and output layers; and $x^{(e)}$ and $o^{(e)}$ correspond to the e-th pair of input and reconstructed signals.

C. Bag representations

The final step of the proposed approach consists in combining local descriptors from a binary image into a single representation for further operations, e.g., retrieval. We do so by constructing bag representations, where each shape is represented by a histogram of the quantization of its local descriptors, which in turn can be estimated by different methods.

In this work in particular, we applied the k-means clustering algorithm to the local descriptors generated by the autoencoder to estimate the so-called visual dictionary. The training set for this, corresponds to the E local descriptors of the same set used to train the autoencoder. Sec. IV-B gives further details of this training step.

III. Data

We used a collection of binary images depicting glyph-blocks, which epigraphers have manually segmented and annotated from scans of three Maya codices (i.e., folded bark-paper books). These codices were produced during the postclassic period of the Maya era (ca. 1100-1520 C.E.), and their importance lies in their extreme rarity as the majority of them were destroyed or are missing, i.e., only these three currently preserved at libraries in Dresden, Paris, and Madrid, are considered of undisputed authenticity [11].

Using the Thompson naming convention [12], which consists of a unique numeric label for each glyph-sign (preceded by the character 'T' for Thompson), i.e., its Thompson number or T-number, every glyph-block in the dataset is annotated by the sequence of Thompson numbers of its constituting glyph-signs, e.g., T0759b-T0025-T0181. These annotations were defined by expert epigraphers following a reading convention defined for Maya hieroglyphs: as indicated in Fig. 1, from left to right, and from top to bottom. Note that, although the sequence of T-numbers suggests already the sequential visual placement of the glyph-signs, there is not certitude of their actual location as they could have been scaled.

TABLE I
MINIMUM, MEAN, AND MAXIMUM NUMBER OF GLYPH-SIGNS IN EACH GLYPH-BLOCK.

min	mean	max
1	2.4±0.7	4

TABLE II
MINIMUM, MEAN, STANDARD DEVIATION, AND MAXIMUM NUMBER OF INSTANCES PER CLASS (71 CLASSES), AND TOTAL NUMBER OF SHAPES.

min	mean±stddev	max	Total
10	27.4±21.9	97	1942

For our experiments, we consider every unique sequence of glyph-signs to be a visual class. For instance, T0759b-T0025-T0181 defines a class of three glyph-signs, and T0024-T1047 another class with only two glyph-signs. Therefore, two sequences of T-numbers, one containing the other, represent different classes, e.g., T0759b-T0025-T0181 is different from T0025-T0181. Fig. 3 shows one example per class in the dataset of Mayan glyph-blocks.

In practice, glyph-blocks in this set are composed by 1 to 4 individual glyph-signs (2.4 on average), at different positions and scales. Table I shows relevant statistics regarding the number of individual signs per glyph-block in the dataset.

As shown in Fig. 3, the dataset consists of 71 visual classes. In turn, each class is composed of 27.4 instances on average. And the complete dataset contains 1942 glyph-blocks. Table II provides further details regarding the distribution of instances per class in the dataset.

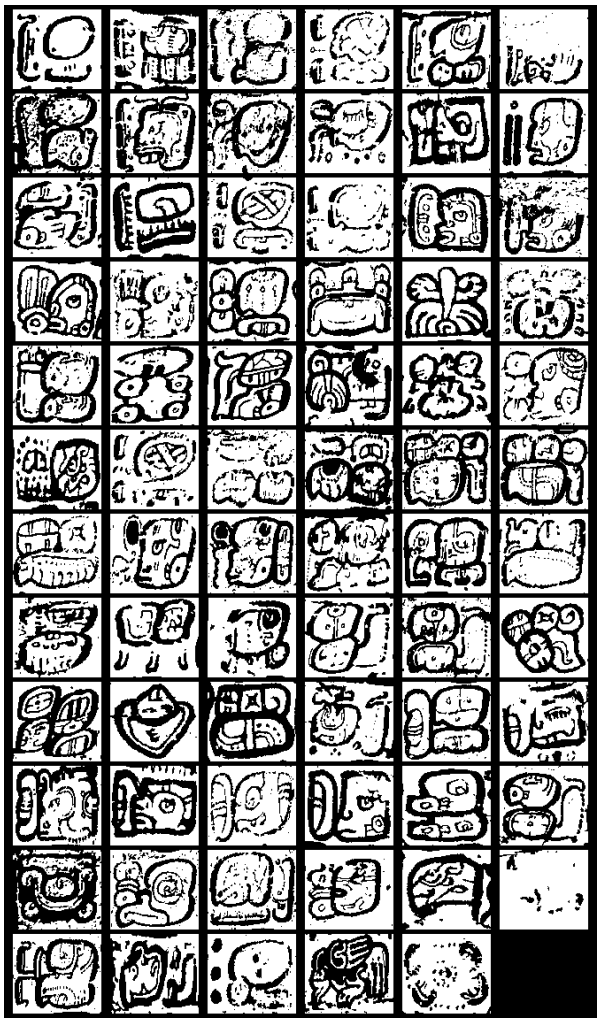

Fig. 3. One visual example per each of the 71 glyph-block classes.

IV. Experimental results

This section presents details regarding the several parameters of the proposed methodology that were adjusted through extensive experimentations.

A. Experimental protocol

To sample points of interest, on which to compute local descriptors, we relied on a common approach [9] [7], in which points are randomly sampled under the constrain of maintaining a uniform spatial distribution over the shape as much as possible. More precisely, for each shape we sampled $0.2M$ points of interest, where M is the total number of points defining the medial axis of the shape of interest.

Also following Belongie's Shape Context approach [9], we defined the locality scope of each point of interest as the average pairwise distance T between all pair of points of interest. This is, the local descriptor for a point of interest corresponds to the set of pixels around it, and not farther than T. Note that the size of the input shapes varies across instances, and so does T and the size of the local patches. Therefore, we resized them to 50x50 pixels before computing the local descriptor. In particular, 50 pixel is an arbitrary value, which through extensive experimentation we found as a good trade-off between size and amount of visual information.

After computing local descriptors using the approach explained in sec. II-B, we computed bag representations (bag-of-words, BoW) using the k-means clustering algorithm. And finally used these bag representations for the retrieval experiments. Namely, we implemented a 5-folds cross-validation approach to estimate visual dictionaries of different sizes, each time feeding k-means with 10,000 local descriptors randomly sampled from the 4 folds used for training. We report our results as the mean average precision computed for the top 10 retrieved instances (mAP@10).

TABLE III
Mean square reconstruction error (MSE) of the autoencoder (local descriptor), with different number of hidden units.

Hidden Units	16	32	64	128	256	512	1024
MSE	0.23	0.23	0.22	**0.11**	0.21	0.22	0.23

TABLE IV
Mean Average Precision (mAP@10) for the first ten retrieved instances, using different approaches for local description.

Dictionary Size	100	250	500	1000	2000
LPAE128	0.325	0.375	**0.379**	0.371	0.364
HOOSC [7]	0.309	0.352	0.368	0.378	**0.396**
LP	0.284	0.288	0.293	0.291	0.289

B. Results

To estimate the appropriate dimensionality for the local descriptor, we evaluated the quality of the reconstructed signal (i.e., local patches) obtained with different number of hidden units in the autoencoder. Table III shows the mean square reconstruction error for the several combinations evaluated.

As seen in Table III, the use of 128 hidden units induces very low reconstruction error. Therefore, we set this value as the dimensionality of the local descriptor.

Similarly, we conducted an exhaustive search to estimate and appropriate dictionary size for bag construction. Namely, a series of retrieval experiments, whose results are reported in Table IV as the mean average precision for the top 10 retrieved instances (mAP@10). The first row of Table IV (LPAE128, after Local Patch and AutoEncoder) corresponds to the proposed approach using 128 hidden units in the autoencoder, as suggested by the results from Table III.

As baseline comparison, we include in the second row of Table IV the performance obtained by the HOOSC descriptor, which has proven robust for description of complex shapes (it has been shown [7] that HOOSC is more robust than the Shape Context descriptor when dealing with complex shapes). More precisely, we used the same set of points of interest sampled for the proposed approach as locations for computing HOOSC descriptors, along with their corresponding local scope defined by T, as explained in sec. IV-A. Consistently with previous

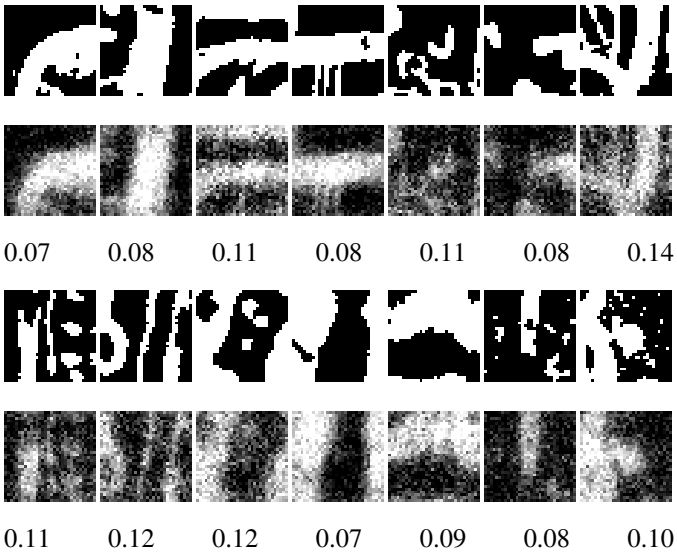

Fig. 4. Local patches randomly sampled and their reconstruction computed mapping back from their corresponding local descriptors. The corresponding MSE is idicated below each pair of input and reconstructed patch.

works, HOOSC performs best with large dictionaries, i.e., 2000 visual words. In contrast, the proposed approach obtains comparable results with a dictionary whose size is only a quarter of that of HOOSC. This is, at the cost of 1.7% of retrieval precision, indexing using the proposed approach can be much more efficient.

A second characteristic that makes the proposed approach desirable over HOOSC, is that, the orientation kernel plus the normalization approaches implemented by HOOSC constitute a series of nonlinear transformations, which make virtually impossible to reconstruct the input local shape. In contrast, the proposed approach relies on a linear autoencoder, which makes easy mapping the local descriptor back to the original input space, thus allowing an approximation of the initial local shape. Note that this is an approximation up to a certain degree, as the information regarding the initial size of the sampled patch is lost after resizing it to 50x50 pixels. Enabling this feature, nevertheless, can be highly relevant for certain scenarios, such as epigraphyc analysis and partial visualization. Fig. 4 shows a set of side-by-side examples of mapping back and forth some local patches.

The third row in Table IV (LP for Local Patch) corresponds to the use of the sampled local patches without the use of the autoencoder. However, these approach showed lower retrieval performance with respect to both LPAE128 and HOOSC.

The curves in Fig. 5 show the average precision as function of the standard recall for the three methods compared in Table IV, computed using bag representations of 500 visual words. Note that the proposed method and HOOSC perform quite close, specially up to 20% of standard recall.

We also compared the performance obtained with a Neural Network trained on the MNIST dataset [13], and with a Convolutional Neural Network trained with the Mayan glyph-

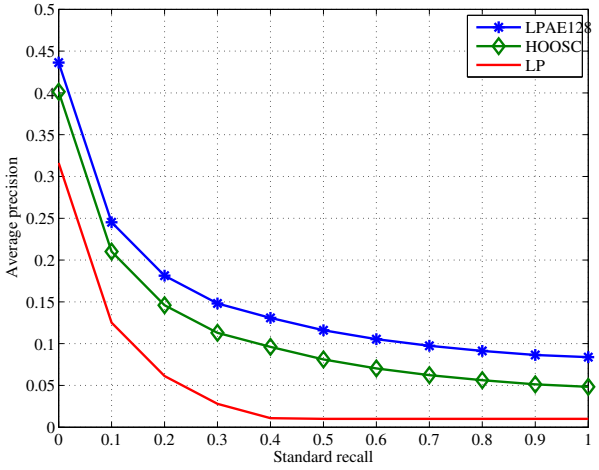

Fig. 5. Average precision vs standard recall curves for the three methods retrieving Mayan glyph-blocks.

blocks, i.e., VGGnet with 16 layers [14]. Namely, after training these two networks to classify shapes, we took the second to last layer and used it as feature vector for indexing the Mayan glyph-blocks. For the NN on MNIST case, the glyph-blocks were resized to fit the input size of the network, thus 28x28 pixels. As consequence, most of the relevant visual information was lost, thus obtaining very poor retrieval performance, i.e., mAP@10 = 0.09, which is about chance.

Regarding the VGGnet, we adjusted all glyph-blocks to fit 224x224 pixels, and repeated the training five times, i.e., once per each of the 5-folds of the glyph-blocks dataset. However, given the large amount of parameters to be learned with respect to the small dataset available for it, only a modest performance of mAP@10=0.12 was obtained.

Finally, Fig. 6 shows visual examples of retrieval experiments conducted using the proposed approach, with the autoencoder of 128 hidden units and a visual dictionary of 500 words. More precisely, glyph-blocks on the first column (with green frame) correspond to queries. Then, the five glyph-blocks identified as most similar to each query are shown in their corresponding row, sorted from left to right in decreasing visual similarity.

V. Conclusion

We presented an approach for computing local descriptors for complex shapes based on autoencoders. This approach is particular useful when only a few examples of data are available for learning shape representations. Specifically, we face the problem of having just a couple of thousand images at 500x700 pixels, which imposes two problems: the input image is too large, thus a large number of parameters must be learned with only a small set of training instances, and down-sampling the images for efficient training neural networks produces inputs which have lose relevant visual information. Namely, initial experiments with such methods produced very low retrieval precision on a dataset of complex shapes.

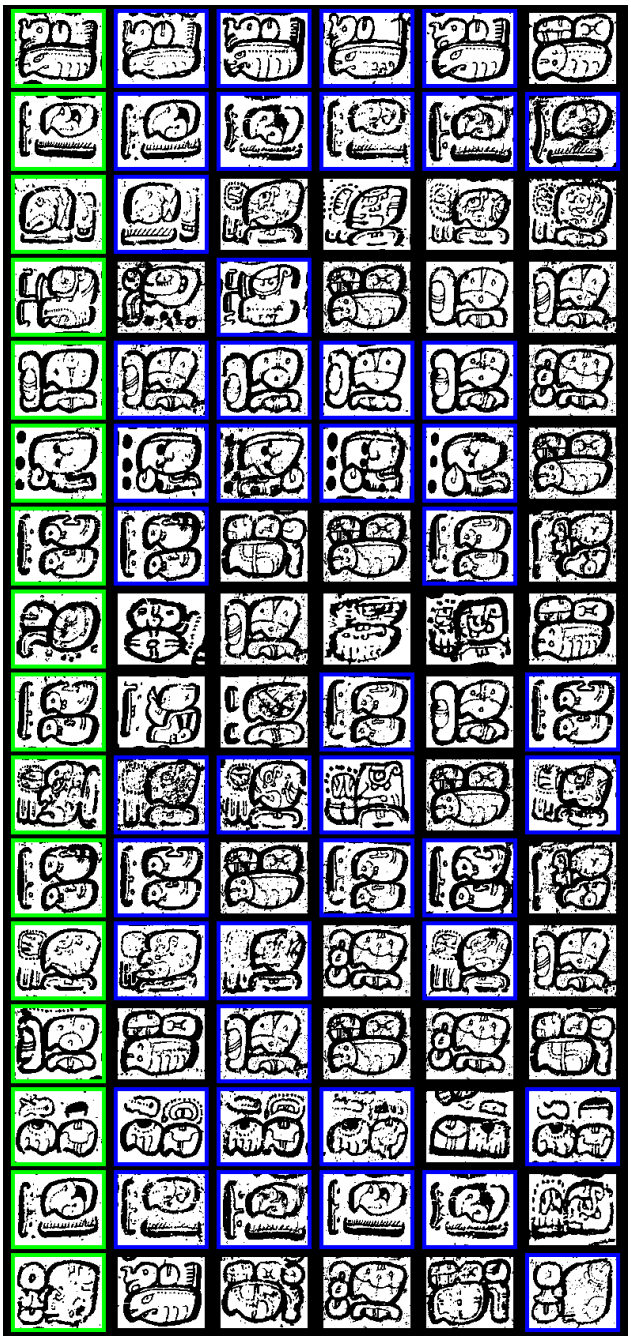

Fig. 6. Visual examples of retrieval results obtained with the proposed approach (128 hidden units and 500 visual words). Glyph-blocks on the first column (with green frame) correspond to queries. The five following glyph-blocks in each row are the most similar ones to each query, sorted from left to right in decreasing visual similarity. Glyph-blocks in a blue frame indicate relevant instances.

In contrast, the proposed approach can deal with this issue by down-sampling only local patches of the images, and learning local representation for them, which later can be combined to construct bag representations. We conducted retrieval experiments of binary images exhibiting very complex shapes of hieroglyphs from the ancient Maya culture. Our results show that our methodology achieves competitive results when compared with handcrafted local shape descriptors, with the advantage that our method also allows for approximating the reconstruction of the input local patch.

Given the results obtained in this work, we plan to direct our efforts towards the automatic extraction of more robust features for improved indexing and retrieval. Namely, given the promise that convolutional networks have for automatically learning robust local representations, we will investigate approaches that would allow us exploiting them for datasets of constrained size.

ACKNOWLEDGMENT

This work was supported by Swiss National Science Foundation through the Maaya project (SNSF - 144238); http://www.idiap.ch/project/maaya/.

REFERENCES

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," in *Intelligent Signal Processing*, 2001.

[3] G. Can, J.-M. Odobez, and D. Gatica-Perez, "Evaluating Shape Representations for Maya Glyph Classification," *ACM Journal on Computing and Cultural Heritage*, vol. V, 2016.

[4] M. Wang, Y. Chen, and X. Wang, "Recognition of Handwritten Characters in Chinese Legal Amounts by Stacked Autoencoders," in *Proceedings of the 22nd International Conference on Pattern Recognition (ICPR)*, 2014.

[5] D. Gatica-Perez, C. P. Gayol, S. Marchand-Maillet, J.-M. Odobez, E. Roman-Rangel, G. Krempel, and N. Grube, "The MAAYA Project: Multimedia Analysis and Access for Documentation and Decipherment of Maya Epigraphy," in *Proceedings of the Digital Humanities Conference (DH)*, 2014.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems*, 2012.

[7] E. Roman-Rangel, C. Pallan, J.-M. Odobez, and D. Gatica-Perez, "Analyzing Ancient Maya Glyph Collections with Contextual Shape Descriptors," *International Journal of Computer Vision (IJCV)*, vol. 94, no. 1, pp. 101–117, 2011.

[8] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning Mid-Level Features for Recognition," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[9] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Transactions on Pattern Analisys Machine Intelligence*, vol. 24, no. 4, pp. 509–522, April 2002.

[10] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, "Deep Learning for Content-Based Image Retrieval: A Comprehensive Study," in *Proceedings of the ACM International Conference on Multimedia*, 2014.

[11] R. Hu, G. Can, C. Pallan-Gayol, G. Krempel, J. Spotak, G. Vail, S. Marchand-Maillet, J.-M. Odobez, and D. Gatica-Perez, "Multimedia Analysis and Access of Ancient Maya Epigraphy," *IEEE Signal Processing Magazine, Special Issue on Signal Processing for Art Investigation*, vol. 32, no. 4, pp. 75–84, July 2015.

[12] J. E. S. Thompson and G. E. Stuart, *A catalog of Maya hieroglyphs*. University of Oklahoma Press Norman, 1962.

[13] A. Krizhevsky and G. E. Hinton, "Using very deep autoencoders for content-based image retrieval," in *19th European Symposium on Artificial Neural Networks (ESANN)*, 2011.

[14] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR*, vol. abs/1409.1556, 2014.