Automatic Egyptian Hieroglyph Recognition by Retrieving Images as Texts

Morris Franken and Jan C. van Gemert
Intelligent Systems Lab Amsterdam (ISLA), University of Amsterdam
Science Park 904 Amsterdam, The Netherlands

ABSTRACT

In this paper we propose an approach for automatically recognizing ancient Egyptian hieroglyph from photographs. To this end we first manually annotated and segmented a large collection of nearly 4,000 hieroglyphs. In our automatic approach we localize and segment each individual hieroglyph, determine the reading order and subsequently evaluate 5 visual descriptors in 3 different matching schemes to evaluate visual hieroglyph recognition. In addition to visual-only cues, we use a corpus of Egyptian texts to learn language models that help re-rank the visual output.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; I.4 [**Image Processing and Computer Vision**]: Features Measurement—*Feature representation, size and shape*

Keywords

Egyptian, Hieroglyphs, Automatic Recognition

1. INTRODUCTION

The ancient Egyptian hieroglyphs have always been a mysterious writing system as their meaning was completely lost in the 4th century AD. The discovery of the Rosetta stone in 1799 allowed researchers to investigate the hieroglyphs, but it wasn't until 1822 when Jean-François Champollion discovered that these hieroglyphs don't resemble a word for each symbol, but each hieroglyph resembles a sound and multiple hieroglyphs form a word. The ability to understand hieroglyphs has uncovered much of the history, customs and culture of Egypt's ancient past.

In this paper we present a system that is able to automatically recognize ancient Egyptian hieroglyphs from photographs. As illustrated in fig 4, a single photograph contains several, often overlapping, hieroglyphs without proper segmentation or a-priori reading order. Moreover, the passing of 4 millennia has introduced noise, and broken or destroyed the original symbols. These conditions present severe challenges to automatic hieroglyph recognition. Automatic hieroglyph recognition is useful for archaeology scholars, the interested amateur, cultural heritage preservation or as a smart-phone App for a tourist or museum visitor.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM'13, October 21–25, 2013, Barcelona, Spain.
Copyright 2013 ACM 978-1-4503-2404-5/13/10 ...$15.00.
http://dx.doi.org/10.1145/2502081.2502199 .

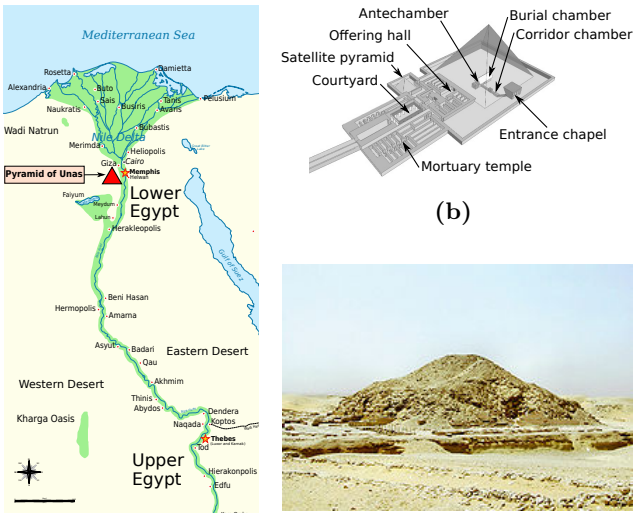

Figure 1: The pyramid of Unas. (a) Location at red triangle. (b) Schematic reconstruction. (c) Current state. *Images courtesy of Wikipedia, creative commons license.*

The paper has 3 contributions. First, we introduce a new hieroglyph dataset where we manually segmented and labeled 3993 hieroglyphs of 10 photographs from the pyramid of Unas. This pyramid is built in the fifth dynasty as a burial place for the Pharaoh *Unas* and is located just south of the city of Giza, see fig 1. We chose a single pyramid to avoid issues with different dialectic writing styles. Second, we show how to automatically locate, segment and recognize hieroglyphs based on visual information. The third contribution is a multimodal extension to the visual analysis with statistical language models from hieroglyph texts. In fig 2 we show a schematic of our approach.

2. RELATED WORK

Multimedia tools have aided preservation, analysis and study of cultural, historical and artistic content. For ex-

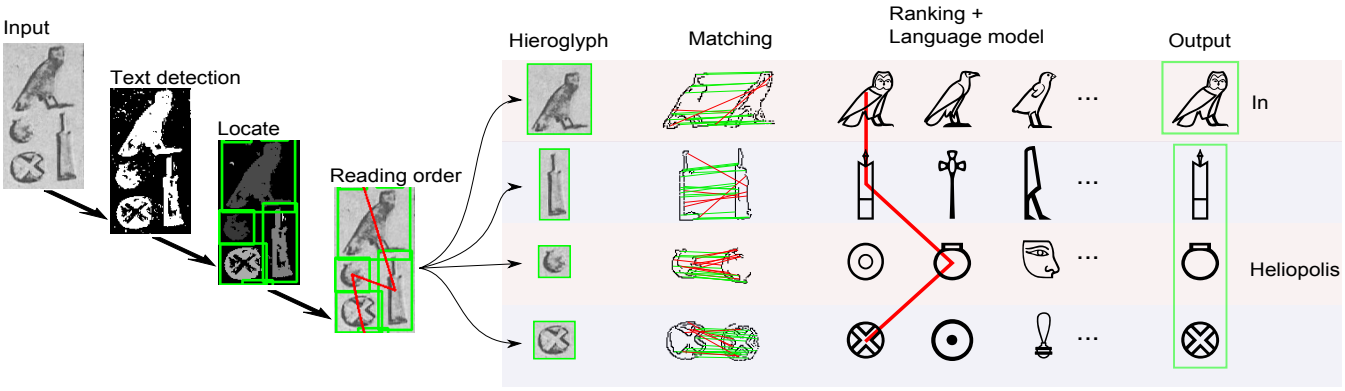

Figure 2: Pipeline for hieroglyph recognition. The 3rd output hieroglyph from the top is corrected by the language model in order to find the word 'Heliopolis' (birth-city of Unas).

ample, the digital Michelangelo project [12] created high quality 3D models of Michelangelo's sculptures and architecture. Furthermore, wavelet analysis of brush strokes in paintings can reveal artist identity [7], image composition has shown to aid category labeling [17] and photographs can be classified as memorable or not [6]. In this paper we follow in these footsteps, and propose a new dataset, and a multimodal (visual and textual) approach for automatic Egyptian hieroglyphs recognition.

Related work on automatic hieroglyph recognition focuses on Mesoamerican culture, and in particular on the ancient Maya hieroglyphs [5, 14, 15]. To this end, the HOOSC descriptor was developed [15], which is a combination of HOG [2] and the Shape-Context [1]. Such descriptors can be used for direct matching [14] or with a bag-of-words (BOW) approach [15]. Other work extracts detailed line segments for Maya hieroglyph matching [5]. In all these works the hieroglyphs are typically manually extracted and individually digitized. In contrast, our photographs consists of noisy plates, which each typically contain around 400 hieroglyphs (see fig 4). Moreover, the Maya culture used a considerable different type of hieroglyphs and we therefore evaluate the HOOSC and other descriptors on Egyptian hieroglyphs.

Current work on automatic scene text detection and recognition [4, 9, 10, 11] are typically hand-tuned to specific western or asian (e.g. Chinese or Hangul) characters which are quite different from Egyptian hieroglyphs. In our work, we will draw inspiration from text detection to localize the hieroglyphs and use generic image descriptors for the recognition.

3. HIEROGLYPH RECOGNITION

Our approach has a visual component where the reading order is determined, hieroglyphs are localized, pre-processed and visually matched. The top-N visual matches are subsequently input to a textual component that re-ranks the hieroglyphs according to a statistical language model trained on other texts. In fig 2 we show a schematic of our approach.

3.1 Localization and Pre-Processing

We localize hieroglyphs with a saliency-based text-detection algorithm [9] (software kindly provided by the authors). This algorithm does not make assumptions on the shape of the text, as done e.g. in [4]. The used algorithm creates a saliency map based on second order curvature statistics and subsequently uses conditional dilation from border-seeds to create a binary text/non-text mask. We group the noisy masks by connected components after a dilation. The output of the localization is an unordered list of bounding boxes (BBs) fitted around each glyph. We experimentally evaluate the localization performance in section 4.

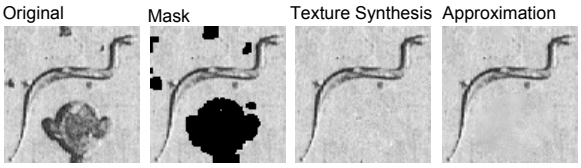

Figure 3: Removing unconnected hieroglyphs.

From the unordered list the reading order is determined to facilitate textual language modeling. The reading order in Egyptian hieroglyphs is either from left to right, right to left or from top to bottom. The only indication of the correct reading order is that glyphs will face the beginning of a line, and top to bottom is indicated by columns separators. Multiple horizontal hieroglyphs in a column should be read as a single line. For the pyramid of Unas the reading order is top-down, from right to left. We sort the unorderd list accordingly to determine the sequence in which the hieroglyphs should be read.

The hieroglyphs BBs are often overlapping or they are in contact with a 'cartouche' (a frame around the name of a royalty). To generate individual hieroglyph images suitable for matching, we filled the overlapping parts with background texture by a non-parametric texture synthesis method [3]. This approach works well, however it is rather slow. We therefore implemented a faster approximation. For each pixel to generate we randomly sample from a search window around the closest filled-in pixel. If the sampled pixel is not part of the foreground mask it is kept, otherwise the process is repeated with a larger window size. After texture synthesis is complete, the final background is smoothed to reduce noise. Our approximation is in the order of 300 times faster, the results of both methods are shown in fig 3. As a final step the patches are extended to 50x75 while retaining their discriminative height/width ratio where the background is again texture synthesized if necessary.

3.2 Image Descriptors

We evaluate five image descriptors, two based on shape, one based on appearance, and the other two on a shape-appearance combination. The Shape-Context (SC) [1] is a shape descriptor originally designed for recognizing handwritten symbols and therefore interesting for hieroglyphs. The SC constructs a spatial log-polar histogram that counts the frequency of edge pixels. Similar to the SC, the Self-Similarity [16] (SS) descriptor was designed for shape matching. Where the SC counts edge pixels, the SS computes the pixel correlation between the central cell to the other cells in a log-polar shaped region. The SC and the SS are shape descriptors, but the hieroglyphs may also benefit from appearance matching. The Histogram of Oriented Gradients (HOG) [2] is a popular appearance descriptor that computes histograms of edge gradient orientations. A combination of shape and appearance can be achieved by merging the HOG and the SC. This combination is called the HOOSC [15] and was developed for Maya hieroglyph matching. The HOOSC replaces the edge pixels in the SC with a HOG. As the fifth descriptor we add a straightforward combination of the SS with HOG which we dub HOOSS. For this combination, the correlations in SS between pixels are replaced with similarities of HOGs. All similarities between K-dimensional descriptors i and j are computed with the χ^2-distance, $\chi^2(i,j) = \sum_1^K \frac{(i_k - j_k)^2}{i_k + j_k}$.

3.3 Visual Matching

To recognize a hieroglyph we compare it to labeled patches in the training set. To this end, we evaluate three common image patch matching schemes. The first method is simply using a single centered descriptor for a full patch. The second approach uses interest points with the popular bag-of-words framework as also used for Maya hieroglyphs [15]. The third approach has also been used for Maya glyphs [14] and uses pair-wise patch matching using interest points with spatial verification. This method starts with a variant of the Hungarian method [8] to assign each descriptor in one patch to a descriptor in the other patch. After obtaining matches, the final matching score s is obtained by fitting an affine transformation between the patches by using Ransac and is computed as $s = m * \sum_{(p_1,p_2)}^{P} \chi^2(p_1,p_2)/|P|^2$, where m is the number of matches, and (p_1, p_2) are matched pairs in the set of Ransac inliers P.

3.4 Language Modeling

The visual ranking is based on a single hieroglyph and does not take the neighboring hieroglyphs into account. The neighborhood can give valuable information since hieroglyphs often co-occur to form words and sentences. To take the context into account we employ language models of hieroglyph occurrences to improve the visual ranking. We compare two strategies: (1) a lexicon lookup and (2) N-grams which represent statistics of N-neighboring hieroglyphs.

The lexicon-based approach tries to match N consequative hieroglyphs to existing words in a dictionary. For each hieroglyph i, we look at the top K=10 results $(v_{i1}, v_{i2}, \ldots, v_{iK})$ of the visual ranking. We re-rank each hieroglyph i by word length $|w|$ and occurrence probability $P(w)$ as $(p(w) + \lambda_w) \cdot |w| \cdot \prod_{j=1}^{K} (v_{i1}/v_{ij})^2$, where w is any exact word match that is present in the Carthesian $N \times K$ space of possible words where N is equal to the largest word in the corpus. To reduce the influence of non-existing words we use a standard Laplace smoothing term λ_w, which we set to 1/20. In section 4 we give the details of the used corpus. We found the best non-linear weighting of visual scores (in this case v^2) on a small hold-out set.

... he shall not write with his little finger. How beautiful is indeed the sight, how good indeed to see, so say they, so say the gods, (when) this god ascends to heaven, (when) Unas ascends to heaven while his power is over him ...

Figure 4: Part of a plate taken from the north wall of the antechamber (column 476) and its translation.

The N-gram approach uses the probability of hieroglyph sub-sequences of length N occurring in a row. We re-rank each hieroglyph i with $\prod_{i=1}^{N} \prod_{j=1}^{K} (v_{i1}/v_{ij})^3 \cdot (p(w) + \lambda_n)$, where w is a hieroglyph sequence of length $N = 3$. To reduce the influence of non-existing N-grams we use a Laplace smoothing term λ_n of 1/2000. Again, we found the best non-linear weighting of visual scores (v^3) on a hold-out set.

4. EXPERIMENTS

We evaluate all descriptors, matching techniques and language models on our new hieroglyph dataset.

4.1 Dataset

The dataset consist of two sets: one being photographs of hieroglyphic texts, the other being the textual corpus that is used for the language model. The visual set consists of 10 plate photographs with hieroglyphs [13], as illustrated in fig 4. These photographs contain 161 unique hieroglyphs with a total of 3993. We manually segmented all individual hieroglyphs with a bounding box and annotated them with their label. To record the label we use a form of transliteration which transforms the hieroglyphs into a character. Many transliteration variations exist among Egyptologists, which are rarely compatible with each other. In this research we chose to use the transliteration used by the open-source JSesh project[1] which also gave rise to the textual set of the database, containing 158 pyramid texts (with a total size of 640KB of text). This textual set is used to train the language models and does not contain any texts from the pyramid of Unas.

4.2 Implementation Details

To reduce differences between descriptors due to parameter setting we keep parameters as equal as possible over the five variants. For HOG we use 8 angle bins, and 4x4 spatial binning. The HOG inside HOOSC and HOOSS also use 8 angle bins. All log-polar histograms have 3 rings, and 8 segments. For the bag-of-words matching we found that a vocabulary size of 200 visual words works well. In the spatial matching we use 500 Ransac iterations. The interest

[1] `http://jsesh.qenherkhopeshef.org/`

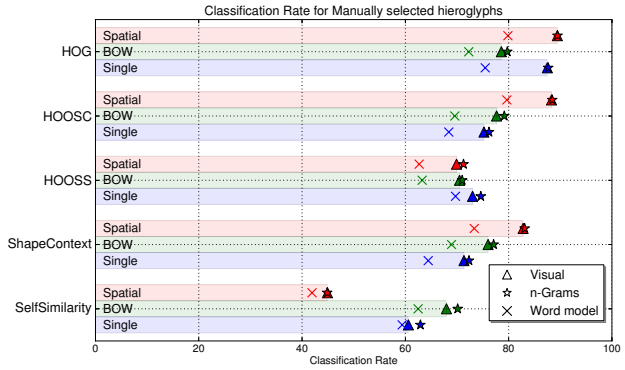

Figure 5: Results for manually cut-out hieroglyphs. The average score is $74 \pm 1\%$.

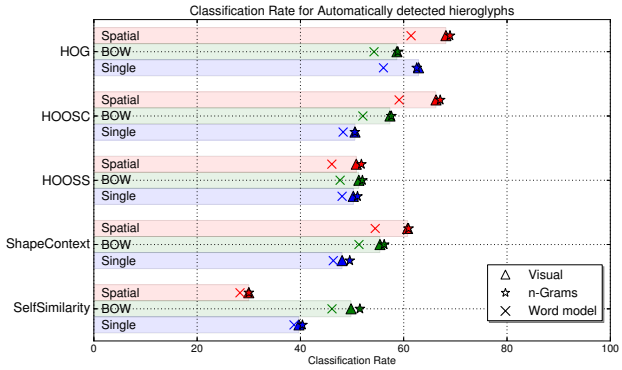

Figure 6: Results for automatically detected hieroglyphs. The average score is 53 ± 5.

points in the BOW and in the spatial matching are based on the contour [14, 15] of a Canny edge detector.

To simulate taking a single photograph, we use one plate for testing and the other plates for training. We repeat the experiment to obtain standard deviations.

4.3 Results

We give results for manually cut-out hieroglyphs recognition in fig 5 and for our automatic detection approach in fig 6. The automatic detection method finds 83% of all manually annotated hieroglyphs, and 85.5% of the detections are correct according to the Pascal VOC overlap criteria. The matching performance trend between the automatic and the manual annotated hieroglyphs is similar, although the single-descriptor HOG seems slightly more sensitive to localization errors.

From the 5 descriptors, the HOOSC and the HOG are the best performers. HOOSC is best on manually annotated hieroglyphs whereas HOG is more robust for automatically detected regions. This seems to indicate that flexibility in spatial structure is important, given the reduced performance of single descriptor HOG on the automatically detected glyphs.

Between the three matching schemes, the spatial-matching performs best. Only for the Self-Similarity the BOW is better. Generally the Self-Similarity does not perform as well on this dataset, which could be attributed to the lack of discriminative features such as color. The slightly better performance of the spatial matching scheme, however, is a factor of 8,000 times slower compared to a single descriptor and 1,000 times slower than the BOW.

Language modeling with a lexicon decreases results on average with 5%. This is due to a bias for smaller words and the lack of word-separators. The N-grams always improves results, albeit slightly, with on average 1%.

5. CONCLUSION

We presented a new Egyptian hieroglyph set with a thorough evaluation of five popular image descriptors, three common matching schemes and two types of language modeling.

6. REFERENCES

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *TPAMI*, 24(4), 2002.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[3] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. In *ICCV*, 1999.

[4] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, pages 2963–2970, 2010.

[5] Y. Frauela, O. Quesadab, and E. Bribiescaa. Detection of a polymorphic mesoamerican symbol using a rule-based approach. *Pattern Recognition*, 39, 2006.

[6] P. Isola, J. Xiao, A. Torralba, and A. Oliva. What makes an image memorable? In *CVPR*, 2011.

[7] R. Johnson, E. Hendriks, J. Berezhnoy, E. Brevdo, S. Hughes, I. Daubechies, J. Li, E. Postma, and J. Wang. Image processing for artist identification. *Signal Processing Magazine, IEEE*, 25(4):37–48, 2008.

[8] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4), 1987.

[9] S. Karaoglu, J. van Gemert, and T. Gevers. Object reading: Text recognition for object recognition. In *ECCV-IFCVCR Workshop*, 2012.

[10] S. Karaoglu, J. van Gemert, and T. Gevers. Con-text: Text detection using background connectivity for fine-grained object classification. In *ACM-MM*, 2013.

[11] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake. Modified quadratic discriminant functions and the application to chinese character recognition. *PAMI*, (1):149–153, 1987.

[12] M. Levoy et al. The digital Michelangelo project: 3D scanning of large statues. In *Computer graphics and interactive techniques*, 2000.

[13] A. Piankoff. *The Pyramid of Unas*. Princeton University Press, 1968.

[14] E. Roman-Rangel, C. Pallan, J.-M. Odobez, and D. Gatica-Perez. Retrieving ancient maya glyphs with shape context. In *ICCV Workshop*, 2009.

[15] E. Roman-Rangel, C. Pallan Gayol, J.-M. Odobez, and D. Gatica-Perez. Searching the past: an improved shape descriptor to retrieve maya hieroglyphs. In *ACM MM*, 2011.

[16] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007.

[17] J. van Gemert. Exploiting photographic style for category-level image classification by generalizing the spatial pyramid. In *ICMR*, 2011.