Viterbi Based Alignment between Text Images and their Transcripts*

Alejandro H. Toselli, Verónica Romero and **Enrique Vidal**
Institut Tecnològic d'Informàtica
Universitat Politècnica de València
Camí de Vera s/n
46071 - València, Spain
[ahector,vromero,evidal]@iti.upv.es

Abstract

An alignment method based on the Viterbi algorithm is proposed to find mappings between word images of a given handwritten document and their respective (ASCII) words on its transcription. The approach takes advantage of the underlying segmentation made by Viterbi decoding in handwritten text recognition based on Hidden Markov Models (HMMs). Two HMMs modelling schemes are evaluated: one using 78-HMMs (one HMM per character class) and other using a unique HMM to model all the characters and another to model blank spaces. According to various metrics used to measure the quality of the alignments, encouraging results are obtained.

1 Introduction

Recently, many on-line digital libraries have been publishing large quantities of digitized ancient handwritten documents, which allows the general public to access this kind of cultural heritage resources. This is a new, comfortable way of consulting and querying this material. The *Biblioteca Valenciana Digital* (BiValDi)[1] is an example of one such digital library, which provides an interesting collection of handwritten documents.

Several of these handwritten documents include both, the handwritten material and its proper transcription (in ASCII format). This fact has motivated the development of methodologies to align these documents and their transcripts; i.e. to generate a mapping between each word image on a document page with its respective ASCII word on its transcript. This word by word alignment would allow users to easily find the place of a word in the manuscript when reading the corresponding transcript. For example, one could display both the handwritten page and the transcript and whenever the mouse is held over a word in the transcript, the corresponding word in the handwritten image would be outlined using a box. In a similar way, whenever the mouse is held over a word in the handwritten image, the corresponding word in the transcript would be highlighted (see figure 1). This kind of alignment can help paleography experts to quickly locate image text while reading a transcript, with useful applications to editing, indexing, etc. In the opposite direction, the alignment can also be useful for people trying to read the image text directly, when arriving to complex or damaged parts of the document.

Creating such alignments is challenging since the transcript is an ASCII text file while the manuscript page is an image. Some recent works address this problem by relying on a previous explicit image-processing based word pre-segmentation of the page image, before attempting the transcription alignments. For example, in (Kornfield et al., 2004), the set of previously segmented word images and their corresponding transcriptions are transformed into two different times series, which are aligned

*This work has been supported by the EC (FEDER), the Spanish MEC under grant TIN2006-15694-C02-01, and by the *Consellería d'Empresa, Universitat i Ciència - Generalitat Valenciana* under contract GV06/252.

[1] http://bv2.gva.es

Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007), pages 9–16,
Prague, 28 June 2007.

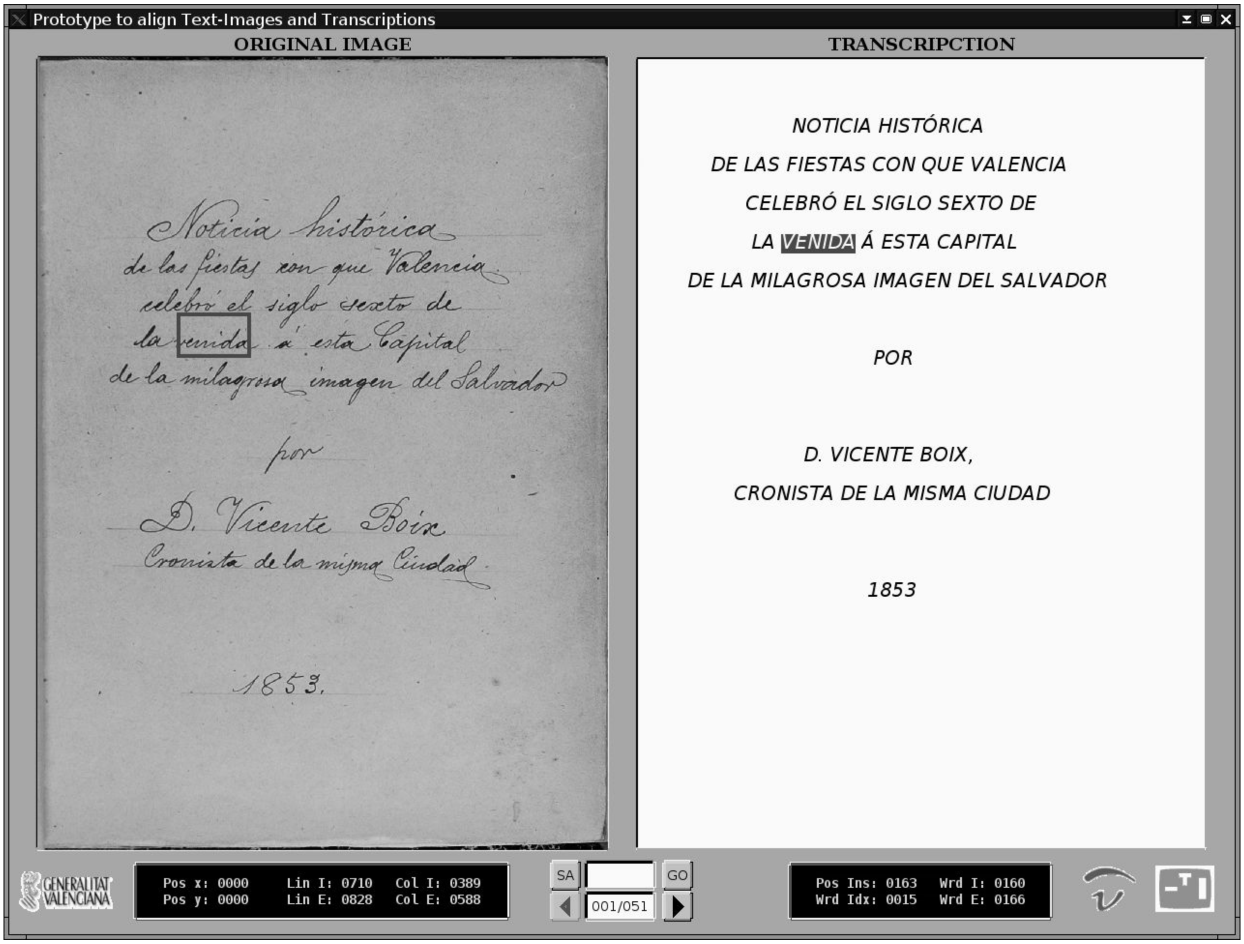

Figure 1: Screen-shot of the alignment prototype interface displaying an outlined word (using a box) in the manuscript (left) and the corresponding highlighted word in the transcript (right).

using *dynamic time warping* (DTW). In this same direction, (Huang and Srihari, 2006), in addition to the word pre-segmentation, attempt a (rough) recognition of the word images. The resulting word string is then aligned with the transcription using dynamic programming.

The alignment method presented here (henceforward called Viterbi alignment), relies on the Viterbi decoding approach to handwritten text recognition (HTR) based on Hidden Markov Models (HMMs) (Bazzi et al., 1999; Toselli et al., 2004). These techniques are based on methods originally introduced for speech recognition (Jelinek, 1998). In such HTR systems, the alignment is actually a byproduct of the proper recognition process, i.e. an implicit segmentation of each text image line is obtained where each segment successively corresponds to one recognized word. In our case, word recognition is not actually needed, as we do already have the correct transcription. Therefore, to obtain the segmentations for the *given* word sequences, the so-called "forced-recognition" approach is employed (see section 2.2). This idea has been previously explored in (Zimmermann and Bunke, 2002).

Alignments can be computed line by line in cases where the beginning and end positions of lines are known or, in a more general case, for whole pages. We show line-by-line results on a set of 53 pages from the *"Cristo-Salvador"* handwritten document (see section 5.2). To evaluate the quality of the obtained alignments, two metrics were used which give information at different alignment levels: one measures the accuracy of alignment mark placements and the other measures the amount of erroneous as-

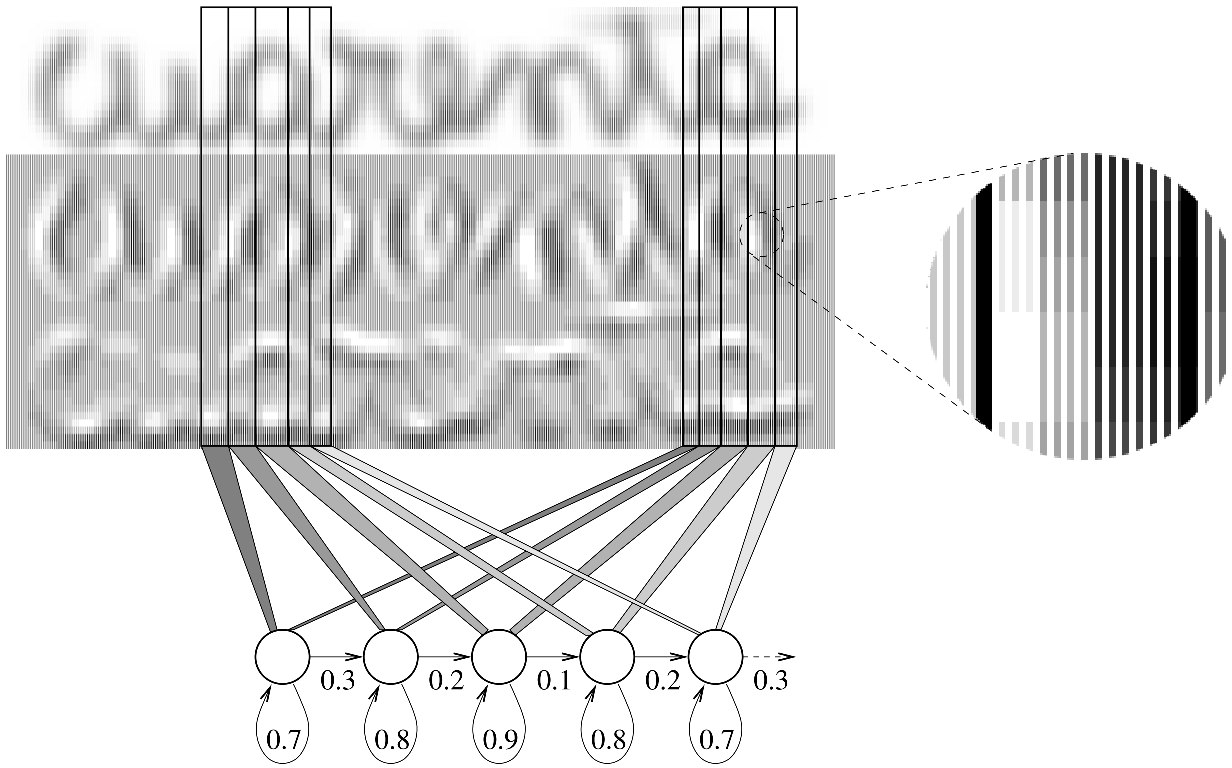

Figure 2: Example of 5-states HMM modeling (feature vectors sequences) of instances of the character “a” within the Spanish word “cuarenta” (forty). The states are shared among all instances of characters of the same class. The zones modelled by each state show graphically subsequences of feature vectors (see details in the magnifying-glass view) compounded by stacking the normalized grey level and its both derivatives features.

signments produced between word images and transcriptions (see section 4).

The remainder of this paper is organized as follows. First, the alignment framework is introduced and formalized in section 2. Then, an implemented prototype is described in section 3. The alignment evaluation metrics are presented in section 4. The experiments and results are commented in section 5. Finally, some conclusions are drawn in section 6.

2 HMM-based HTR and Viterbi alignment

HMM-based handwritten text recognition is briefly outlined in this section, followed by a more detailed presentation of the Viterbi alignment approach.

2.1 HMM HTR Basics

The traditional handwritten text recognition problem can be formulated as the problem of finding a most likely word sequence $\hat{\mathbf{w}} = \langle w_1, w_2, \ldots, w_n \rangle$, for a given handwritten sentence (or line) image represented by a feature vector sequence $\mathbf{x} = x_1^p = \langle x_1, x_2, \ldots, x_p \rangle$, that is:

$$\begin{aligned} \hat{\mathbf{w}} &= \arg\max_{\mathbf{w}} Pr(\mathbf{w}|\mathbf{x}) \\ &= \arg\max_{\mathbf{w}} Pr(\mathbf{x}|\mathbf{w}) \cdot Pr(\mathbf{w}) \end{aligned} \quad (1)$$

where $Pr(\mathbf{x}|\mathbf{w})$ is usually approximated by concatenated character Hidden Markov Models (HMMs) (Jelinek, 1998; Bazzi et al., 1999), whereas $Pr(\mathbf{w})$ is approximated typically by an n-gram word language model (Jelinek, 1998).

Thus, each character class is modeled by a continuous density left-to-right HMM, characterized by a set of states and a Gaussian mixture per state. The Gaussian mixture serves as a probabilistic law to model the emission of feature vectors by each HMM state. Figure 2 shows an example of how a HMM models a feature vector sequence corresponding to

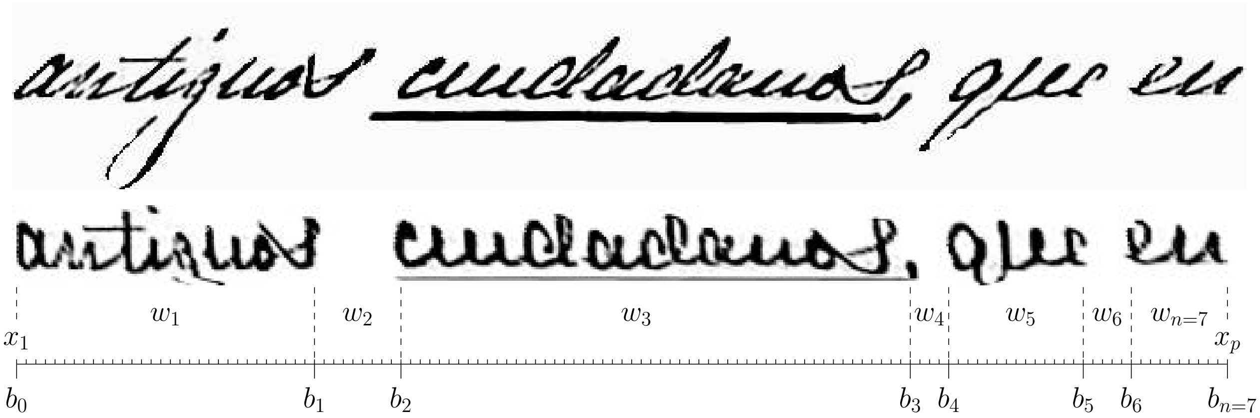

Figure 3: Example of segmented text line image along with its resulting deslanted and size-normalized image. Moreover, the alignment marks ($b_0 \dots b_8$) which delimit each of the words (including word-spaces) over the text image feature vectors sequence $\mathbf{x}$.

character "a". The process to obtain feature vector sequences from text images as well as the training of HMMs are explained in section 3.

HMMs as well as n-grams models can be represented by stochastic finite state networks (SFN), which are integrated into a single global SFN by replacing each word character of the n-gram model by the corresponding HMM. The search involved in the equation (1) to decode the input feature vectors sequence $\mathbf{x}$ into the more likely output word sequence $\hat{\mathbf{w}}$, is performed over this global SFN. This search problem is adequately solved by the Viterbi algorithm (Jelinek, 1998).

2.2 Viterbi Alignment

As a byproduct of the Viterbi solution to (1), the feature vectors subsequences of $\mathbf{x}$ aligned with each of the recognized words $w_1, w_2, \dots, w_n$ can be obtained. These implicit subsequences can be visualized into the equation (1) as follows:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \sum_{\mathbf{b}} Pr(\mathbf{x}, \mathbf{b}|\mathbf{w}) \cdot Pr(\mathbf{w}) \quad (2)$$

where $\mathbf{b}$ is an *alignment*; that is, an ordered sequence of $n+1$ marks $\langle b_0, b_1, \dots, b_n \rangle$, used to demarcate the subsequences belonging to each recognized word. The marks b_0 and b_n always point out to the first and last components of $\mathbf{x}$ (see figure 3).

Now, approximating the sum in (2) by the dominant term:

$$\hat{\mathbf{w}} \approx \arg\max_{\mathbf{w}} \max_{\mathbf{b}} Pr(\mathbf{x}, \mathbf{b}|\mathbf{w}) \cdot Pr(\mathbf{w}) \quad (3)$$

where $\hat{\mathbf{b}}$ is the optimal alignment. In our case, we are not really interested in proper text recognition because the transcription is known beforehand. Let $\tilde{\mathbf{w}}$ be the given transcription. Now, $Pr(\mathbf{w})$ in equation 3 is zero for all $\mathbf{w}$ except $\tilde{\mathbf{w}}$, for which $Pr(\tilde{\mathbf{w}}) = 1$. Therefore,

$$\hat{\mathbf{b}} = \arg\max_{\mathbf{b}} Pr(\mathbf{x}, \mathbf{b}|\tilde{\mathbf{w}}) \quad (4)$$

which can be expanded to,

$$\hat{\mathbf{b}} = \arg\max_{\mathbf{b}} Pr(\mathbf{x}, b_1|\tilde{\mathbf{w}}) Pr(\mathbf{x}, b_2|b_1, \tilde{\mathbf{w}}) \dots \\ \dots Pr(\mathbf{x}, b_n|b_1 b_2 \dots b_{n-1}, \tilde{\mathbf{w}}) \quad (5)$$

Assuming independence of each b_i mark from $b_1 b_2 \dots b_{i-1}$ and assuming that each subsequence $x_{b_{i-1}}^{b_i}$ depends only of $\tilde{w}_i$, equation (5) can be rewritten as,

$$\hat{\mathbf{b}} = \arg\max_{\mathbf{b}} Pr(x_{b_0}^{b_1}|\tilde{w}_1) \dots Pr(x_{b_{n-1}}^{b_n}|\tilde{w}_n) \quad (6)$$

This simpler Viterbi search problem is known as "forced recognition".

3 Overview of the Alignment Prototype

The implementation of the alignment prototype involved four different parts: document image preprocessing, line image feature extraction, HMMs training and alignment map generation.

Document image preprocessing encompasses the following steps: first, skew correction is carried out on each document page image; then background removal and noise reduction is performed by applying a bi-dimensional median filter (Kavallieratou and Stamatatos, 2006) on the whole page image. Next, a text line extraction process based on local minimums of the horizontal projection profile of page image, divides the page into separate line images (Marti and Bunke, 2001). In addition connected components has been used to solve the situations where local minimum values are greater than zero, making impossible to obtain a clear text line separation. Finally, slant correction and non-linear size normalization are applied (Toselli et al., 2004; Romero et al., 2006) on each extracted line image. An example of extracted text line image is shown in the top panel of figure 3, along with the resulting deslanted and size-normalized image. Note how non-linear normalization leads to reduced sizes of ascenders and descenders, as well as to a thiner underline of the word "ciudadanos".

As our alignment prototype is based on Hidden Markov Models (HMMs), each preprocessed line image is represented as a sequence of feature vectors. To do this, the feature extraction module applies a grid to divide line image into $N \times M$ squared cells. In this work, $N = 40$ is chosen empirically (using the corpus described further on) and M must satisfy the condition $M/N =$ *original image aspect ratio*. From each cell, three features are calculated: normalized gray level, horizontal gray level derivative and vertical gray level derivative. The way these three features are determined is described in (Toselli et al., 2004). Columns of cells or *frames* are processed from left to right and a feature vector is constructed for each *frame* by stacking the three features computed in its constituent cells.

Hence, at the end of this process, a sequence of M 120-dimensional feature vectors (40 normalized gray-level components, 40 horizontal and 40 vertical derivatives components) is obtained. An example of feature vectors sequence, representing an image of the Spanish word *"cuarenta"* (forty) is shown in figure 2.

As it was explained in section 2.1, characters are modeled by continuous density left-to-right HMMs with 6 states and 64 Gaussian mixture components per state. This topology (number of HMM states and Gaussian densities per state) was determined by tuning empirically the system on the corpus described in section 5.1. Once a HMM "*topology*" has been adopted, the model parameters can be easily trained from images of continuously handwritten text (*without any kind of segmentation*) accompanied by the transcription of these images into the corresponding sequence of characters. This training process is carried out using a well known instance of the EM algorithm called *forward-backward or Baum-Welch re-estimation* (Jelinek, 1998).

The last phase in the alignment process is the generation of the mapping proper by means of Viterbi "forced recognition", as discussed in section 2.2.

4 Alignment Evaluation Metrics

Two kinds of measures have been adopted to evaluate the quality of alignments. On the one hand, the average value and standard deviation (henceforward called MEAN-STD) of the absolute differences between the system-proposed word alignment marks and their corresponding (correct) references. This gives us an idea of the geometrical accuracy of the alignments obtained. On the other hand, the alignment error rate (AER), which measures the amount of erroneous assignments produced between word images and transcriptions.

Given a reference mark sequence $\mathbf{r} = \langle r_0, r_1, \ldots, r_n \rangle$ along with an associated tokens sequence $\mathbf{w} = \langle w_1, w_2, \ldots, w_n \rangle$, and a segmentation marks sequence $\mathbf{b} = \langle b_0, b_1, \ldots, b_n \rangle$ (with $r_0 = b_0 \wedge r_n = b_n$), we define the MEAN-STD and AER metrics as follows:

MEAN-STD: The average value and standard deviation of absolute differences between reference and proposed alignment marks, are given by:

$$\mu = \frac{\sum_{i=1}^{n-1} d_i}{n-1} \quad \sigma = \sqrt{\frac{\sum_{i=1}^{n-1} (d_i - \mu)^2}{n-1}} \tag{7}$$

where $d_i = |r_i - b_i|$.

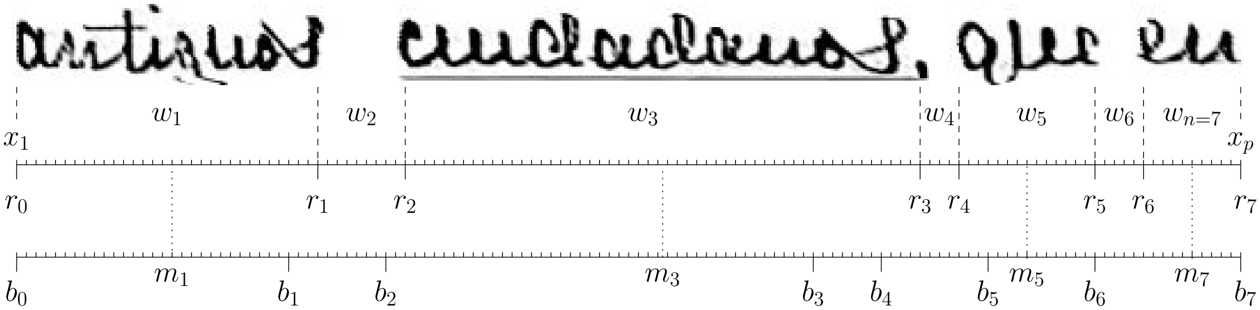

Figure 4: Example of AER computation. In this case $N = 4$ (only no word-space are considered: w_1, w_3, w_5, w_7) and w_5 is erroneously aligned with the subsequence $x_{b_5}^{b_6}$ ($m_5 \notin (b_4, b_5)$). The resulting AER is 25%.

AER: Defined as:

$$\text{AER}(\%) = \frac{100}{N} \sum_{j:w_j \neq b} e_j \qquad e_j = \begin{cases} 0 & b_{j-1} < m_j < b_j \\ 1 & \text{otherwise} \end{cases} \tag{8}$$

where b stands for the blank-space token, $N < n$ is the number of real words (i.e., tokens which are not b, and $m_j = (r_{j-1} + r_j)/2$.

A good alignment will have a μ value close to 0 and small σ. Thus, MEAN-STD gives us an idea of how accurate are the automatically computed alignment marks. On the other hand, AER assesses alignments at a higher level; that is, it measures mismatches between word-images and ASCII transcriptions (tokens), excluding word-space tokens. This is illustrated in figure 4, where the AER would be 25%.

5 Experiments

In order to test the effectiveness of the presented alignment approach, different experiments were carried out. The corpus used, as well as the experiments carried out and the obtained results, are reported in the following subsections.

5.1 Corpus description

The corpus was compiled from the legacy handwriting document identified as *Cristo-Salvador*, which was kindly provided by the *Biblioteca Valenciana Digital* (BIVALDI). It is composed of 53 text page images, scanned at 300dpi and written by only one writer. Some of these page images are shown in the figure 5.

As has been explained in section 3, the page images have been preprocessed and divided into lines, resulting in a data-set of 1,172 text line images. In this phase, around 4% of the automatically extracted line-separation marks were manually corrected. The transcriptions corresponding to each line image are also available, containing 10,911 running words with a vocabulary of 3,408 different words.

To test the quality of the computed alignments, 12 pages were randomly chosen from the whole corpus pages to be used as references. For these pages the true locations of alignment marks were set manually.

Table 1 summarized the basic statistics of this corpus and its reference pages.

Number of:	References	Total	Lexicon
pages	12	53	–
text lines	312	1,172	–
words	2,955	10,911	3,408
characters	16,893	62,159	78

Table 1: Basic statistics of the database

5.2 Experiments and Results

As mentioned above, experiments were carried out computing the alignments line-by-line. Two different HMM modeling schemes were employed. The first one models each of the 78 character classes using a different HMM per class. The second scheme uses 2 HMMs, one to model all the 77 no-blank character classes, and the other to model only the blank "character" class. The HMM topology was identical for all HMMs in both schemes: left-to-right with 6 states and 64 Gaussian mixture com-

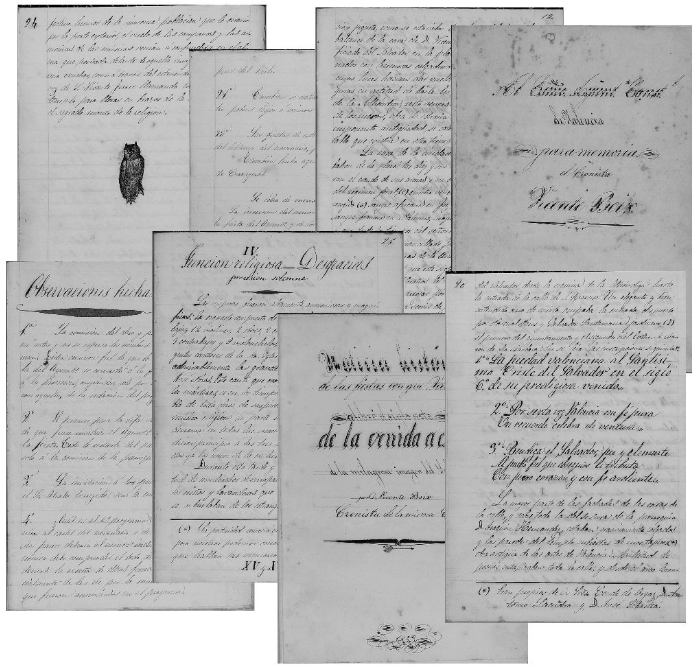

Figure 5: Examples page images of the corpus "Cristo-Salvador", which show backgrounds of big variations and uneven illumination, spots due to the humidity, marks resulting from the ink that goes through the paper (called bleed-through), etc.

ponents per state.

As has been explained in section 4, two different measures have been adopted to evaluate the quality of the obtained alignments: the MEAN-STD and the AER. Table 2 shows the different alignment evaluation results obtained for the different schemes of HMM modeling.

	78-HMMs	2-HMMs
AER (%)	7.20	25.98
μ (mm)	1.15	2.95
σ (mm)	3.90	6.56

Table 2: Alignment evaluation results 78-HMMs and 2-HMMs.

From the results we can see that using the 78 HMMs scheme the best AER is obtained (7.20%). Moreover, the relative low values of μ and σ (in millimeters) show that the quality of the obtained alignments (marks) is quite acceptable, that is they are very close to their respective references. This is illustrated on the left histogram of figure 6.

The two typical alignment errors are known as over-segmentation and under-segmentation respectively. The over-segmentation error is when one word image is separated into two or more fragments. The under-segmentation error occurs when two or more images are grouped together and returned as one word. Figure 7 shows some of them.

6 Remarks and Conclusions

Given a manuscript and its transcription, we propose an alignment method to map every word image on the manuscript with its respective ASCII word on the transcript. This method takes advantage of the implicit alignment made by Viterbi decoding used in text recognition with HMMs.

The results reported in the last section should be considered preliminary.

Current work is under way to apply this alignment approach to the whole pages, which represents a more general case where the most corpora do not have transcriptions set at line level.

References

I. Bazzi, R. Schwartz, and J. Makhoul. 1999. An Omnifont Open-Vocabulary OCR System for English and

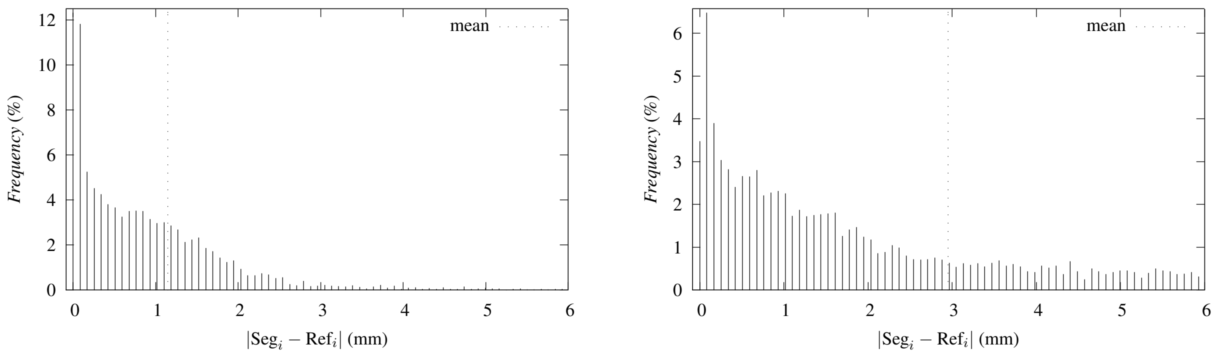

Figure 6: $|r_i - b_i|$ distribution histograms for 78-HMMs (left) and 2-HMMs (right) modelling schemes.

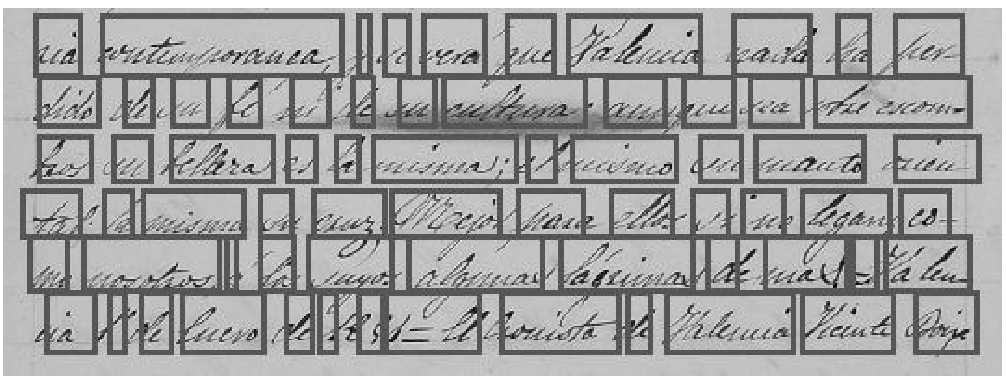

Figure 7: Word alignment for 6 lines of a particularly noisy part of the corpus. The four last words on the second line as well as the last line illustrate some of over-segmentation and under-segmentation error types.

Arabic. *IEEE Trans. on PAMI*, 21(6):495–504.

Chen Huang and Sargur N. Srihari. 2006. Mapping Transcripts to Handwritten Text. In Suvisoft Ltd., editor, *Tenth International Workshop on Frontiers in Handwriting Recognition*, pages 15–20, La Baule, France, October.

F. Jelinek. 1998. *Statistical Methods for Speech Recognition*. MIT Press.

Ergina Kavallieratou and Efstathios Stamatatos. 2006. Improving the quality of degraded document images. In *DIAL '06: Proceedings of the Second International Conference on Document Image Analysis for Libraries (DIAL'06)*, pages 340–349, Washington, DC, USA. IEEE Computer Society.

E. M. Kornfield, R. Manmatha, and J. Allan. 2004. Text Alignment with Handwritten Documents. In *First International Workshop on Document Image Analysis for Libraries (DIAL)*, pages 195–209, Palo Alto, CA, USA, January.

U.-V. Marti and H. Bunke. 2001. Using a Statistical Language Model to improve the preformance of an HMM-Based Cursive Handwriting Recognition System. *Int. Journal of Pattern Recognition and Artificial In telligence*, 15(1):65–90.

V. Romero, M. Pastor, A. H. Toselli, and E. Vidal. 2006. Criteria for handwritten off-line text size normalization. In *Procc. of The Sixth IASTED international Conference on Visualization, Imaging, and Image Processing (VIIP 06)*, Palma de Mallorca, Spain, August.

A. H. Toselli, A. Juan, D. Keysers, J. Gonzlez, I. Salvador, H. Ney, E. Vidal, and F. Casacuberta. 2004. Integrated Handwriting Recognition and Interpretation using Finite-State Models. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 18(4):519–539, June.

M. Zimmermann and H. Bunke. 2002. Automatic Segmentation of the IAM Off-Line Database for Handwritten English Text. In *ICPR '02: Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 4*, page 40035, Washington, DC, USA. IEEE Computer Society.