# Overview of the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task

Mauricio Villegas, Joan Puigcerver, Alejandro H. Toselli,
Joan-Andreu Sánchez and Enrique Vidal

PRHLT, Universitat Politècnica de València
Camí de Vera s/n, 46022 València, Spain
{mauvilsa,joapuipe,ahector,jandreu,evidal}@prhlt.upv.es

**Abstract.** The ImageCLEF 2016 Handwritten Scanned Document Retrieval Task was the first edition of a challenge aimed at developing retrieval systems for handwritten documents. Several novelties were introduced in comparison to other recent related evaluations, specifically: multiple word queries, finding local blocks of text, results in transition between consecutive pages, handling words broken between lines, words unseen in training and queries with zero relevant results. To evaluate the systems, a dataset of manuscripts written by Jeremy Bentham was used, and has been left publicly available after the evaluation. The participation was not as good as expected, receiving results from four groups. Despite the low participation, the results were very interesting. One group obtained very good performance, handling relatively well the cases of queries with words not observed in the training data and locating words broken between two lines.

## 1 Introduction

In recent years there has been an increasing interest in digitizing the vast amounts of pre-digital age books and documents that exist throughout the world. Many of the emerging digitizing initiatives are aimed at dealing with huge collections of handwritten documents, for which automatic recognition is not yet as mature as for printed text Optical Character Recognition (OCR). Thus, there is a need to develop reliable and scalable indexing techniques for manuscripts, targeting its particular challenges. Users for this technology could be libraries with fragile historical books, which for preservation are being scanned to make them available to the public without the risk of further deterioration. Apart from making the scanned pages available, there is also interest in providing search facilities so that the people consulting these collections have information access tools that they are already accustomed to have in other applications. Another use for this technology is historical birth, marriage and death records, due to the common interest of people in finding out about their ancestors and family roots, which results in a need for tools that ease searching these collections of records.

The archaic solution to handwritten document indexing is to manually transcribe and then use standard text retrieval technologies. However, this becomes too expensive for large collections and more so for historical documents that generally require expert paleographers to transcribe them. Alternatively, handwritten text recognition (HTR) techniques can be used for automatic indexing, which requires to transcribe only a small part of the document for learning the models, or reuse models obtained from similar manuscripts, thus requiring the least human effort. The use of HTR for indexing manuscripts does not have to be limited to only recognizing the text before the use of standard text retrieval techniques. The uncertainty of the recognized text can be taken into account during the retrieval, so the HTR can potentially provide better performance than what standard text retrieval techniques would allow.

This paper presents an overview of the Handwritten Scanned Document Retrieval challenge, one of the three evaluation tasks organized by ImageCLEF in 2016 under the CLEF initiative labs[1]. The main aspects that distinguishes this challenge in comparison to other evaluations or competitions related to searching in handwritten documents are: multiple word queries, finding local blocks of text, results in transition between consecutive pages, handling words broken between lines, words unseen in training and queries with zero relevant results. The reminder of this paper is organized as follows. Next section gives a short overview of works related to this challenge. Then, Section 3 describes the task in detail, including the objective of the challenge, the reasons why it was organized, participation rules and provided data and resources. Followed by this, Section 4 presents and discusses the results of the systems submitted by the participants. Section 5 gives recommendations for using the dataset from this evaluation in future works. Finally, Section 6 concludes the paper with final remarks and future outlooks. For a global view of the ImageCLEF activities in 2016, the reader is invited to look at the general overview paper [20].

## 2 Related Work

Traditionally, the process of automatically searching for text in handwritten documents has been known as *Keyword Spotting* (KWS), which actually can be seen as a particular case of image retrieval. The goal of KWS is to find all instances of a query word in a given document. Among the noteworthy KWS paradigms aiming to fulfill this goal, two main kinds are distinguished: *Query-by-Example* (QbE) [8,7,1,23] and *Query-by-String* (QbS) [4,5,15,19]. While in QbE the query is specified by providing a cropped image of the word or words to be searched for, in QbS, queries are directly specified as character strings, i.e., typing in a keyboard. Likewise other distinctions considered are: training-based or training-free [4,23]; i.e., whether the KWS system needs or not to be trained on appropriate (annotated) images, and segmentation-based or segmentation-free [23,7]; i.e., whether KWS is applied to full document (page) images or just to images of individual words (previously segmented from the original full images). Finally, it is also worth mentioning that, similar as most state-of-the-art handwritten text recognition systems, there are KWS approaches which are line-oriented [4,5,19].

[1] http://www.clef-initiative.eu

This means that their basic input search/indexing units are whole text line images, without any kind of segmentation into words or characters, which are analyzed to determine the degree of confidence that a given keyword appears in the image.

Both of the paradigms, QbE and QbS, have their particular usage scenarios, strengths and weaknesses. In the case of QbE, the techniques generally are training-free, so they alleviate the need to transcribe a part of the document. However, they do have the limitation that the query is an image, which depending on the case might not be easy to obtain. In contrast, QbS techniques can potentially allow to search for any word, although they generally require training.

In recent years, several KWS contests on handwritten documents have been organized, mainly conjunction with conferences like the International Conference on Frontiers in Handwriting Recognition (ICFHR) and the International Conference on Document Analysis and Recognition (ICDAR). These contests focused first on evaluating QbE approaches [11][2,3], although lately, QbS approaches have been also considered in the ICDAR'15 [12][4], and in the ICFHR'16[5].

# 3 Overview of the Task

## 3.1 Motivations and Objectives

The general motivations for this handwritten retrieval task have already been outlined in the introduction of the paper, so there is no need to repeat them. Though, additional to these there are other more specific motivations related the current state of research in this area and the other evaluation initiatives that have been organized in recent years.

Keyword spotting is mostly concerned with detecting all instances of a given word in a set of images, and with good reason since any system for searching text in images containing handwriting must be based on a technique of this sort. However, since this is a problem related to searching, it must also be analyzed from the perspective of the field of information retrieval. There are other difficulties beyond the detections of words that are important for the development of handwritten retrieval systems, so one objective of this evaluation was to stimulate the research in other aspects of the problem that are also important. One aspect that was considered was the type of query the user issues. In order to favor ease of use in general applications, a good approach is to allow queries written as natural language, in contrast to for example allowing only a single word for searching, or multiple words as a boolean expression. With the idea of targeting this goal in the future, for this evaluation the queries were composed of multiple words and the order of the words had to be taken into account.

[2] http://users.iit.demokritos.gr/~bgat/H-WSCO2013

[3] http://vc.ee.duth.gr/H-KWS2014

[4] http://transcriptorium.eu/~icdar15kws

[5] https://www.prhlt.upv.es/contests/icfhr2016-kws

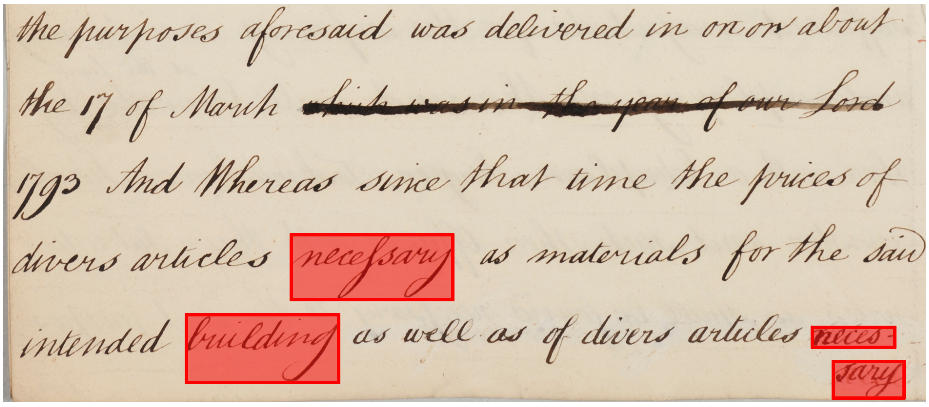
Another aspect considered was what were the objects to retrieve. Previous research on keyword spotting has mostly targeted finding a word in a line of text or in a full page. A single line is generally too small for multiple word queries. On the other hand, providing a full page as result might not be convenient for the user to have a quick look (i.e., read part of the text retrieved) and judge if the result does fulfill the current information need. For some applications a middle ground between a line and a page might be more appropriate. Furthermore, handwritten documents are not composed of isolated pages, since the content traverses multiple pages. The result of a query could lie in a transition between two consecutive pages, so retrieval systems should consider this possibility. A related topic is that commonly when someone writes, when reaching the end of a line, if there is little space to fit completely the next word, it is split or broken between the current line and the following one. In many cases these broken words are marked with a symbol, such as the hyphen (short horizontal line) commonly used now, but in past times other symbols have been employed. Also there are documents in which there is no special mark for the broken words, either because it was forgotten or because it was not a usual practice. These broken words are also candidates to be found, so developing techniques to handle them is also a requirement.

As mentioned before, these kinds of evaluations related to handwriting recognition are normally organized in conjunction with conferences such as ICDAR and ICFHR, which are not common venues for the information retrieval community. This was one of the reasons for organizing it at CLEF, to present these problems to experts on information retrieval and foster collaborations that surely will benefit this area of research.

### 3.2 Challenge Description

The challenge aimed at evaluating the performance of retrieval systems for scanned handwritten documents[6]. Specifically, it targeted the scenario of free text search in a document, in which the user wants to find locations for a given multiple word query provided as a string (QbS). The result of the search are local regions (such as a paragraph), which could even include the end of a page and start of the next. However, since layout analysis and detection of paragraphs is in itself a difficult problem, a couple of simplifications were introduced. First, the was no requirement to detect the text lines in the page images or determine their reading order. This was provided as part of the data. Second, the segments to retrieve were defined as a concatenation of 6 consecutive lines (from top to bottom and left to right if there are columns), ignoring the type of line it may be (e.g. title, inserted word, etc.). More precisely, the segments were defined by a sliding window that moves one line at a time (thus neighboring segments overlap by 5 lines) traversing all the pages in the document, so there are segments that include lines at the end of a page and at the beginning of the next, and the total

[6] Challenge website at http://imageclef.org/2016/handwritten



the purposes aforesaid was delivered in on or about
the 17 of March ~~which was in the year of our Lord~~
1793 And Whereas since that time the prices of
divers articles neceſsary as materials for the said
intended building as well as of divers articles neces-
sary

**Fig. 1:** Example of a 6-line segment extracted from the development set. This segment is relevant for the query *building necessary*, whose word appearances are highlighted in the image. The second appearance of *necessary* is broken between two lines.

number of segments is five less than the total number of lines in the document. An example of a segment is presented in Figure 1.

The queries selected for evaluation consisted of between one and five words that had to be searched for in the collection of pages, and a segment was considered relevant if all the query words appeared in the given order. Due to the overlap of the segments, for a given query, several consecutive segments are relevant. In a real application these consecutive segments would be considered a single result, however, to simplify the evaluation, the relevancy of all segments had to be provided by the systems. The participants were expected to submit for each query, only for the segments considered relevant, a relevancy score and the bounding boxes of all appearances of the query words within the segment, irrespectively if it was or not an instance of the word that made the segment relevant. To clarify this last point, taking the example from Figure 1, the word *necessary* appears twice, however, the segment is relevant only because of the second appearance, since the order of the words in the query has to be the same. Nevertheless, the bounding box of first appearance of *necessary* also had to be provided.

The queries were selected such that some key challenges were included, so the developed systems were expected to consider them. These challenges are:

1. Queries with zero relevant results. In these cases the systems are expected to assign low relevancy scores. Due to the uncertainty of the recognized text, this would allow to apply a threshold to filter low confidence recognitions and prevent showing false positives to the users.
2. Queries with results relevant due to words broken between two lines. See Figure 1 for an example. Both parts of the broken word have to be within the six lines of the segment. Not all broken words in the dataset include a hy-

phenation symbol. Also the two parts of a broken word are not necessarily in consecutive lines since there can be inserted text between these two lines.

3. Queries including words not seen in the data used for learning the recognition models, also known as out-of-vocabulary (OOV) words.
4. Queries with a repeated word, so this word had to appear as many times in the segment as it did in the query, additional to the restriction of the word order.

Since the evaluation measured the impact of words not seen in the training set, the use of external data for learning a language model was prohibited. The use of external data for learning the optical models was allowed, but with the condition that results were also submitted using the same techniques and only the provided data for learning the optical models. Each registered group was allowed to submit a maximum of 10 system runs. If a group chose to submit results using external training data, an additional 10 systems runs were allowed, to cover the ones with and without external training data.

A development set with accompanying ground truth was provided so that participants could tune their systems. However, it was prohibited to use this development data as training for the final submissions, since the participants were required to submit retrieval results for the concatenation of the development and test sets, as if it were a single large document.

The task was designed to allow easy participation from different research communities by providing prepared data for each (see subsection 3.3), with the aim of having synergies between these communities, and providing different ideas and solutions to the problems being addressed. Not only groups that work on handwritten text recognition could participate. For researchers specialized on text processing and retrieval, recognition results in plain text using a baseline system were provided. Researchers that worked in query-by-example could also participate, since for the training set, bounding boxes for the words were given, thus allowing to obtain example images for a given query string. However, this kind of participation had twist, that the training words were segmented automatically, therefore could be incorrectly segmented. Thus a technique to select the among the available example words would be required.

### 3.3 Dataset

The dataset used in this task was a subset of pages from unpublished manuscripts written by the philosopher and reformer, Jeremy Bentham, that have been digitised and transcribed under the Transcribe Bentham project [3]. Refer to Table 1 for some of the dataset statistics. The data was divided into three sets: training with 363 pages, development with 433 pages and test with 200 pages. The participants had to submit results for the concatenation of development and test as if it were a single 633 page document. For the three sets, the original scanned color page images were made available so that all parts of the challenge could be addressed, including extraction of lines, pre-processing of the images, training of recognition models, decoding, indexing and retrieval. To ease participation, also

**Table 1:** Statistics for the Bentham dataset and queries used in the evaluation.

| | **Training** | **Devel.** | **Test** | **Test 99** |
|---|---|---|---|---|
| Pages | 363 | 433 | 200 | 99 |
| Segments | - | 10,584 | 6,350 | 2,972 |
| Lines | 9,645 | 10,589 | 6,355 | 3,021 |
| Running words[a] | 75,132 | 91,346 | Unk.[b] | 20,686 |
| Total queries | - | 510 | 1,000 | 1,000 |
| Unique words in queries | - | 1,055 | 1,643 | 1,643 |
| Queries with OOV words | - | 178 | 425 | 425 |
| Queried broken words | - | 186 | Unk.[b] | 192 |
| Relevant segments | - | 10,367 | Unk.[b] | 3,493 |
| Rel. segm. for OOV queries[c] | - | 1,268 | Unk.[b] | 1,083 |
| Rel. segm. with broken words[d] | - | 736 | Unk.[b] | 1,032 |

[a] Without tokenizing.
[b] Unknown since 101 pages had not been transcribed at the time.
[c] Relevant segments only for the queries with OOV words.
[d] Relevant segments with a broken word as its only appearance.

pre-extracted line images were provided. These lines had in their meta-data the crop window offset required to compute bounding boxes in the original page image coordinates. The identifiers of the lines were set to be incrementing integers such that the identifier of the first line in a 6-line segment coincided with the number of the corresponding segment. For the training and development sets, the transcripts were also provided, in the first case to allow training recognition models and in the second so that participants could also define new queries to evaluate in the development set.

For each line of the development and test sets, the 100-best recognitions using the baseline system (see subsection 3.4) were given, including log-likelihoods and word bounding boxes. These n-best recognitions were intended for researchers that worked on related fields (such as text retrieval) but do not normally work with images or on handwriting recognition, so that they could easily participate, concentrating on other parts of the challenge.

For each page image there was also an accompanying XML in Page 2013 format[7]. The training set XMLs included polygons surrounding every line that had been manually checked, and also word polygons but in this case automatically obtained. The automatic word polygons were intended for groups working in query-by-example keyword spotting, although the corresponding word bounding boxes were provided separately for convenience. In contrast to training, the development set XMLs had manually checked polygons for both lines and words and the test set only had manually checked baselines instead of polygons surrounding the lines. Thus, for the test set, the participants additionally needed to

[7] http://www.primaresearch.org/tools/PAGELibraries

develop a technique to crop the lines for a given baseline, or use the pre-extracted line images.

For the development set, a list of 510 queries and the respective retrieval ground truth (obtained using the word polygons) was supplied. The list of queries for the test set had the same ones from development plus 490 additional, thus 1,000 in total. These 1,000 queries included 1,643 unique words.

The 200 test set pages were kind of special, since the procedure to select them and obtain its ground truth was significantly different with respect to the development and training sets. The development set was actually the first batch of images processed in the tranScritprium project [17], and were chosen to be a consecutive set of very homogeneous pages with not so many evident difficulties. The training set was the second batch considered in tranScritprium, which was also consecutive and homogeneous but somewhat more challenging than the first one. For the test set, the pages were selected among ones that were transcribed by volunteers [2], so there was no guarantee that the pages were consecutive. The actual number of transcribed pages from the test set was not the full 200 pages, but only 99. Many of the others were included since they were pages the between the transcribed ones. To measure the performance for the test set, the segments from untranscribed pages were removed from the submissions. However, since the retrieval results are available, the ground truth could be produced for these missing pages so that the performance can be obtained for the full set.

All of the provided data for the task and scripts for computing the evaluation measures and the baseline system (see subsections 3.4 and 3.5) are now publicly available and citable [21].

### 3.4 Baseline System

A very simple baseline system was provided so that it served as a starting point or just for initial comparisons and understanding the submission format. Since the evaluation was designed to allow participation from researchers working on text retrieval, the baseline system was chosen so that it served as a basis for this kind of participation. Thus, the baseline system can be divided into two disjoint modules: 1) the first module receives the line images and produces a list of the 100 most probable recognitions according to a pre-trained HTR model; 2) then the second module takes as input this list of recognitions and estimates the probability that the query words appear in the same order for any given 6-line segment. Only for the second module of the system, software was provided to the participants so that they could reproduce the results for the development set using the 100-best recognitions.

The first module of the system was based on the classical architecture composed of three main processes: document image pre-processing, line image feature extraction and Hidden Markov Model (HMM) and language model training/decoding [18]. The pre-processing included: extraction of line images, correction of line slope and slant, noise removal and image enhancement [22]. Feature vectors were extracted using the technique form [9]. Training of HMMs and decoding was done using HTK[8] and bi-gram language model trained using SRLIM[9]. The HMM character models had 6 states, Gaussian Mixture Models (GMM) of 64 components and were trained for 5 Expectation Maximization (EM) iterations. The HVite decoder was used to generate the 100-best recognition lists, using input degree of 15, Grammar Scale Factor (GSF) of 15 and Word Insertion Penalty (WIP) of -5. These parameters were selected based on previous works on the Bentham data.

Once the 100-best recognition lists are built for each line image, a similar approach to the one described in [13] is followed to compute the probability that a given segment, comprised by image lines $\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+L-1}$, is relevant for a particular query, $\mathbf{q}$. Following the previous work, this probability can be computed by the following expression:

$$P(R = 1 \mid \mathbf{q}, \mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+L-1}) = \tag{1}$$

$$\sum_{\mathbf{w} \in \Sigma^*} P(R = 1, \mathbf{w} \mid \mathbf{q}, \mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+L-1}) = \tag{2}$$

$$\sum_{\mathbf{w}' \in \Sigma^* : R(\mathbf{w}', \mathbf{q}) = 1} P(\mathbf{w}' \mid \mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+L-1}) \tag{3}$$

$R$ is a binary random variable that denotes whether or not the segment is relevant for the query $\mathbf{q}$. The first probability distribution is marginalized across all possible transcripts, $\mathbf{w}$, of the segment. This is equivalent to the probability of all segment transcripts, $\mathbf{w}'$, that are relevant for the query $\mathbf{q}$, denoted by the binary function $R(\mathbf{w}', \mathbf{q})$. Character-Lattices were used in [13] to compute the previous probabilities at a line-level. Here, the 100-best line-level transcripts were used instead to approximate $P(\mathbf{w} \mid \mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+L-1})$.

Observe that using a 100-best list of line-level transcription hypotheses yields to a much bigger set of segment-level hypotheses. More precisely, with segments comprised by $L$ lines, if $N$-best line-level hypotheses are used, this gives a total number of $N^L$ segment-level hypotheses. In our setting, with $L = 6$ and $N = 100$, that would result in $10^{12}$ hypotheses for each segment. Instead, 100 segment-level hypotheses were formed by joining first the 1-best hypotheses from all lines, then the 2-best hypotheses, and so on. This strategy is clearly suboptimal, but enables working with a much smaller set of segment-level hypotheses and is enough for a reasonable baseline system. The log-likelihood of the segment-level transcripts are computed as the sum of the log-likelihoods of the line-level transcripts.

In order to determine which transcript hypotheses are relevant for a particular query, a regular expression containing all the words in the query, optionally separated by other words, is created and then the likelihoods of all segment-level hypotheses matching this regular expression are added and the result is divided by the total likelihood of the segment to obtain the probability described in Eq. (1). Regarding the word-level bounding boxes, they are extracted from the bounding boxes of the transcript hypothesis containing the query that has the highest log-likelihood.

[8] http://htk.eng.cam.ac.uk

[9] http://www.speech.sri.com/projects/srilm/

The baseline system has two main shortcomings that were left for the participants to tackle and improve: the presence of OOV words in the queries and broken words split into two lines. Python source code[10] was made available for the participants to have all the details of the baseline system. Also a script that computes the performance measures described next was made available[11] along with the development set baseline retrieval results and ground truth, so that it could be tested and the measures compared with the ones published in the challenge web page.

### 3.5 Performance Measures

Two well known information retrieval metrics, Average Precision (AP) and Normalized Discounted Cumulative Gain (NDCG), were taken as basis to benchmark the systems, but applied in several ways so that the different challenges considered in the evaluation were measured. These metrics were computed in two ways: 1) globally and 2) for each query individually and then taking the mean. To distinguish these two ways of measuring performance, we prepend to the acronyms the letter $g$ for global and $m$ for the mean. Thus, we have gAP as the AP measured globally (in the literature commonly referred to as just AP [14]) and the mAP as the mean of the APs obtained for each of the queries. Likewise, we have gNDCG as the NDCG applied globally and mNDCG as the mean for all queries (commonly referred to as just NDCG in the literature).

The gAP and gNDCG are highly affected by the queries with zero relevant results, due to the fact that all queries are considered at once, thus comparing the relevancy scores across queries. The systems that give low relevancy scores for queries without relevants are rewarded because true relevants from other queries will be ranked better than for the zero relevant queries.

For two other novel challenges we decided to directly measure its performance: broken words and OOV words. In the case of OOV words, we just measured the performance only for the queries that included at least one word that was not observed in the training data. To measure the performance for the broken words, we restricted further the requirement for a segment to be relevant, such that it contains at least one broken word, and that the broken word is the only appearance of this word in the segment.

Since the objects to retrieve are parts of images containing (possibly difficult to read) handwriting, it is important for the systems to provide locations of the words to serve as a visual aid for the users. This is why it was required that participants provide bounding boxes of the spotted words. Even though the word locations are important, the measure of retrieval performance should be independent of the detection of words, so the four metrics were computed both at segment and word bounding box levels.

All performance metrics aim to measure how good are the results given by a ranked list of $N$ matches, given that there are $R$ relevant elements expected to

[10] https://zenodo.org/record/52994/files/nbest-baseline.py

[11] https://zenodo.org/record/52994/files/assessment.py

be retrieved, according to the ground-truth. The metrics can be defined in terms of two fundamental concepts: *true positive* (correct matches) and *false positives* (also known as type I errors), which are used to classify a particular retrieved match. Let $\mathrm{TP}(k)$ and $\mathrm{FP}(k)$ be indicator functions that evaluate to 1 if the $k$-th match is a true positive or a false positive, respectively, otherwise they evaluate to 0. Then, the precision $p(k)$ among the first $k$ elements of the list, is defined as:

$$p(k) = \frac{\sum_{j=1}^{k} \mathrm{TP}(j)}{\sum_{j=1}^{k} \mathrm{TP}(j)+\mathrm{FP}(j)} \tag{4}$$

The usual definition of AP only considers the cases when $N > 0$ (there is at least one match in the retrieved list) and $R > 0$ (there is at least one relevant element in the ground-truth). For this evaluation, the definition of AP was extended to address these ill-defined cases as follows:

$$\mathrm{AP} = \begin{cases} \frac{1}{R} \sum_{k=1}^{N} p(k) \cdot \mathrm{TP}(k) & N > 0 \wedge R > 0 \\ 1.0 & N = 0 \wedge R = 0 \\ 0.0 & \text{otherwise} \end{cases} \tag{5}$$

The gAP is computed using Eq. (5) for all queries combined, and the mAP is defined as:

$$\mathrm{mAP} = \frac{1}{|Q|} \sum_{q \in Q} \mathrm{AP}(q) \tag{6}$$

where $\mathrm{AP}(q)$ corresponds to Eq. (5) computed only for query $q$, and $Q$ is the set of all queries being evaluated.

Similar to the AP, the NDCG was extended to address the ill-defined cases as follows:

$$\mathrm{NDCG} = \begin{cases} \frac{1}{Z} \sum_{k=1}^{N} \frac{2^{\mathrm{TP}(k)} - 1}{\log_2(k+1)} & N > 0 \wedge R > 0 \\ 1.0 & N = 0 \wedge R = 0 \\ 0.0 & \text{otherwise} \end{cases} \tag{7}$$

where $Z$ is a normalization factor (sometimes denoted as INDCG), computed from the ideal retrieval result, so that the NDCG is a value between 0.0 and 1.0. Analogous to the AP, the gNDCG is computed using Eq. (7) for all queries combined, and the mNDCG is defined as

$$\mathrm{mNDCG} = \frac{1}{|Q|} \sum_{q \in Q} \mathrm{NDCG}(q) \tag{8}$$

where $\mathrm{NDCG}(q)$ corresponds to Eq. (7) computed only for query $q$, and $Q$ is the set of all queries being evaluated.

To measure the previous performance measures at a word level, the TP and FP were generalized to a continuous case by using the intersection over

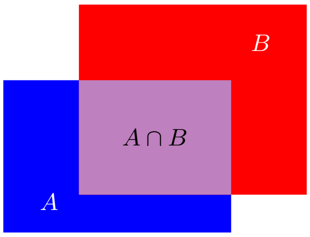


**Fig. 2:** The blue rectangle ($A$) is a detected bounding box and the red rectangle ($B$) is a reference bounding box. Then, the area of the intersection, $A \cap B$, over the total area of both bounding boxes is the *true positive* (TP) fraction of the match, the remaining area of the box $A$ over the total area of $A$ is the *false positive* (FP) fraction, i.e. the relative fraction of the reference box that was not matched.

union (IoU) metric as an overlapping degree between the reference and the retrieved bounding boxes. Let $A$ be a detected bounding box in the retrieved list of matches, and $B$ a reference bounding box corresponding to the same word, then the continuous definitions of TP and FP are:

$$\mathrm{TP}(A,B) = \frac{|A \cap B|}{|A \cup B|} = \mathrm{IoU} \tag{9}$$

$$\mathrm{FP}(A,B) = \frac{|A| - |A \cap B|}{|A|} = 1 - \frac{|A \cap B|}{|A|} \tag{10}$$

These continuous versions can be more easily understood by the visual example depicted in Fig. 2.

Given these extensions, the bounding box of the $k$-th result in the retrieved list is matched against the previously-unmatched reference bounding box (of the same word) with the greatest IoU area. That is, a reference bounding box which was not matched against any of the previous $k-1$ results. This prevents that the same reference bounding box is matched multiple times against different detected bounding boxes. Then, the continuous versions of TP and FP are applied in the definitions of the four metrics described above.

## 4 Evaluation Outcome

### 4.1 Participation

There was considerable interest in the task. In total 48 groups registered, and 24 of these signed the End User Agreement (a requirement to access the data during the evaluation). Based on the data server logs, the test queries (only useful if there was some intention of submitting results) were downloaded from 9 countries: Germany, India, Israel, Japan, Mexico, Morocco, Tunisia, Turkey and USA. In the end, only four groups submitted results and three of them submitted a working notes paper describing their system. Among the four groups, 37 system runs were submitted, 7 of which had mistakes thus were invalidated.

The final participation was not as good as hoped for, though considering that it was a novel challenge and it was organized in conjunction with an unusual venue for its topic, the participation was not bad.

The following is a list of short descriptions, one for each of the participating groups (ordered alphabetically). The reference of the paper describing their systems, for the three groups that submitted it, is included below so that the reader can refer to them for further details of the used techniques.

- **CITlab:** [16] The team from the Computational Intelligence Technology Laboratory of the Institute of Mathematics, University of Rostock, in Germany. It was represented by Tobias Strauß, Tobias Grüning, Gundram Leifert and Roger Labahn. They were the only team that participated in the full challenge, training their own recognition models and dealing with broken words and out-of-vocabulary words. The technique is based on multi-dimensional recurrent neural networks (MDRNN) trained using connectionist temporal classification (CTC), followed by a regular expression based decoder aided by a word uni-gram language model. To handle broken words, when lines are recognized with a hyphenation symbol at the end or the start, the respective parts are concatenated to check if it matches as a word. The CITlab group submitted 18 runs in total, among which 8 corresponded to systems that where trained with external training data. After the submission, they realized that in the external training data, there were pages from the development set, which was prohibited in the evaluation. So they decided to exclude these results in their working notes paper, and they are excluded in this paper because of this also.
- **MayoBMI:** [10] The team from the Section of Biomedical Informatics of the Mayo Clinic, in Rochester, United States of America. It was represented by Sijia Liu, Yanshan Wang, Saeed Mehrabi, Dingcheng Li and Hongfang Liu. They focused their research on the detection of hyphenation symbols followed by a spell correction to detect broken words, but finally did not submit results using this development. The text retrieval of their system was based on the n-best recognition results provided by the organizers, though only considering the 1-best. To index the words they used the Porter stemmer and the retrieval was based on Term Frequency - Inverse Document Frequency (TF-IDF). The MayoBMI group submitted only one system run.
- **IIIT:** The team from the Center for Visual Information Technology, International Institute of Information Technology, in Hyderabad, India. It was represented by Kartik Dutta. This was the only participant that followed the query-by-example route. Interestingly, their submission handled out-of-vocabulary words, but unfortunately they did not submit a paper describing the approach and decided not to disclose any details about it. The IIIT group submitted two system runs.
- **UAEMex:** [6] The team from the Universidad Autónoma del Estado de México, (UAEM), México. It was represented by Miguel Ángel García Calderón and René Arnulfo García Hernández. Their method was based on the n-best recognition results provided by the organizers, though only considering the 1-best. The technique was based on the Longest Common Subsequence al-

gorithm, which made it possible to retrieve words not seen in training. The UAEMex group submitted 8 system runs.

### 4.2 Results

The complete results of the evaluation are presented in six tables. The first two tables, 2 and 3, correspond to the complete retrieval results. The Following two, 4 and 5, are for the additional requirement of the segments containing broken words. And the final two tables, 6 and 7, are using only the queries that include at least one OOV word.

All of these tables include the results for both the development and the 99 page test sets, and for the four performance measures described in subsection 3.5. There is no special order for the results within the tables. The groups appear in alphabetical order and the runs for each group correspond to the order in which they were submitted. The best result from each group is highlighted in bold font. This selection is based on considering equally important the eight computed values (development, test and 4 performance measures). The cells in the tables where there is a dash as result, are the cases in which the system run did not include even a single retrieval result. This indicates that the system run had some problem or did not consider that specific challenge, for example the case of broken and OOV words for the baseline.

### 4.3 Discussion

Each group followed quite a different path. The IIIT team participated as query-by-example, thus their results are not directly comparable with any of the other participants. Hopefully in the future other research groups will use this dataset for the query-by-example scenario, and be able to compare with what is presented in this paper. Two teams, MayoBMI and UAEMex, based their work on the provided recognition results, although considered only the 1-best hypothesis, so they are handicapped in comparison to the baseline system. Furthermore, the test set was considerably more difficult than the development and the baseline system performed very poorly, so their results were also affected by this. Two groups, CITlab and MayoBMI, dealt with the broken words, though both based it on the detection of hyphenation symbols, so the cases in which there is no hyphenation are ignored. CITlab and UAEMex proposed solutions that handled OOV words.

The performance obtained by the CITlab is very impressive. They used recurrent neural networks, which is the current state-of-the-art in handwriting recognition, so the good performance was somewhat expected. Many groups in this area are working on similar techniques, so the low participation was unfortunate since it prevented a much more fruitful comparison of systems. The results for the OOV queries are not affected much in comparison to the complete set of queries. This suggests that the problem of the OOV can be handled rather well with current techniques. In the case of the broken words, the drop in performance is more noticeable, though it can be said that it is a more challenging problem

**Table 2:** Results (in %) for the segment-based performance measures and the complete retrieval results.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | 74.2 | 14.4 | 49.9 | 8.1 | 80.1 | 27.5 | 51.7 | 9.4 |
| CITlab #1 | 95.4 | 43.7 | 89.8 | 40.0 | 96.8 | 62.1 | 90.8 | 41.9 |
| CITlab #2 | 94.8 | 42.2 | 89.4 | 38.2 | 96.7 | 61.1 | 90.6 | 40.2 |
| CITlab #3 | 91.7 | 36.2 | 88.6 | 36.5 | 96.3 | 59.7 | 90.0 | 38.8 |
| CITlab #4 | 92.6 | 39.6 | 89.1 | 37.9 | 96.5 | 60.6 | 90.4 | 40.0 |
| CITlab #5 | 94.8 | 33.9 | 89.7 | 38.6 | 96.7 | 59.4 | 90.8 | 40.6 |
| CITlab #6 | 95.4 | 43.2 | 89.6 | 39.1 | 96.8 | 61.4 | 90.7 | 41.0 |
| CITlab #7 | 95.0 | 46.7 | 89.6 | 39.2 | 96.7 | 62.0 | 90.7 | 41.0 |
| CITlab #8 | 91.9 | 36.3 | 88.9 | 37.1 | 96.4 | 60.4 | 90.3 | 39.5 |
| CITlab #9 | 94.9 | 34.0 | 89.9 | 39.5 | 96.8 | 60.0 | 91.0 | 41.4 |
| **CITlab #10** | **95.0** | **47.1** | **89.8** | **39.9** | **96.8** | **62.7** | **90.9** | **41.7** |
| **IIIT #1** | **41.5** | **3.4** | **22.5** | **3.4** | **49.4** | **8.8** | **26.1** | **3.9** |
| IIIT #2 | 41.6 | 1.2 | 22.5 | 2.5 | 49.4 | 3.9 | 26.1 | 2.7 |
| **MayoBMI #1** | **25.8** | **2.5** | **23.4** | **2.9** | **33.1** | **7.0** | **26.6** | **3.6** |
| **UAEMex #1** | **61.1** | **0.3** | **38.5** | **0.4** | **69.1** | **1.2** | **41.7** | **0.4** |
| UAEMex #2 | 47.6 | 0.0 | 32.3 | 0.0 | 59.4 | 0.0 | 37.6 | 0.0 |
| UAEMex #3 | 30.2 | 0.0 | 20.3 | 0.0 | 43.6 | 0.0 | 27.1 | 0.0 |
| UAEMex #4 | - | - | - | - | - | - | - | - |
| UAEMex #5 | 51.2 | 3.5 | 36.9 | 0.9 | 64.6 | 10.2 | 40.7 | 1.5 |
| UAEMex #6 | 27.6 | 0.0 | 19.8 | 0.0 | 53.8 | 0.0 | 28.9 | 0.0 |
| UAEMex #7 | 0.1 | 0.0 | 1.7 | 0.0 | 1.9 | 0.0 | 2.8 | 0.0 |
| UAEMex #8 | 26.2 | 0.0 | 19.6 | 0.0 | 53.4 | 0.0 | 28.8 | 0.0 |

**Table 3:** Results (in %) for the box-based performance measures and the complete retrieval results.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | 53.1 | 6.2 | 41.3 | 4.9 | 51.6 | 10.8 | 38.6 | 5.3 |
| **CITlab #1** | **72.9** | **22.5** | **72.9** | **28.2** | **75.4** | **36.4** | **76.0** | **31.6** |
| CITlab #2 | 69.6 | 23.1 | 72.0 | 26.4 | 74.5 | 36.2 | 75.5 | 30.0 |
| CITlab #3 | 66.7 | 21.3 | 71.5 | 25.2 | 73.8 | 35.7 | 75.0 | 28.9 |
| CITlab #4 | 68.5 | 23.2 | 71.8 | 26.3 | 74.2 | 36.4 | 75.2 | 29.9 |
| CITlab #5 | 72.7 | 17.8 | 72.9 | 27.2 | 75.4 | 34.6 | 75.9 | 30.6 |
| CITlab #6 | 73.4 | 23.0 | 72.9 | 27.6 | 75.6 | 36.2 | 76.0 | 31.0 |
| CITlab #7 | 70.9 | 25.6 | 72.4 | 27.4 | 74.7 | 36.7 | 75.7 | 30.8 |
| CITlab #8 | 66.9 | 20.5 | 71.5 | 25.9 | 73.9 | 35.9 | 75.0 | 29.5 |
| CITlab #9 | 72.3 | 17.0 | 72.9 | 27.7 | 75.2 | 34.6 | 76.0 | 31.1 |
| CITlab #10 | 70.5 | 25.1 | 72.3 | 28.1 | 74.4 | 37.1 | 75.6 | 31.5 |
| **IIIT #1** | **20.1** | **0.0** | **18.8** | **0.0** | **24.2** | **0.0** | **21.1** | **0.0** |
| IIIT #2 | 20.1 | 0.0 | 18.8 | 0.0 | 24.2 | 0.0 | 21.1 | 0.0 |
| **MayoBMI #1** | **18.4** | **1.0** | **18.4** | **1.7** | **25.7** | **3.4** | **22.2** | **2.5** |
| UAEMex #1 | 7.2 | 0.1 | 8.7 | 0.1 | 17.8 | 0.4 | 14.1 | 0.1 |
| **UAEMex #2** | **13.1** | **0.0** | **14.7** | **0.0** | **26.2** | **0.0** | **21.2** | **0.0** |
| UAEMex #3 | 6.2 | 0.0 | 8.4 | 0.0 | 16.6 | 0.0 | 14.2 | 0.0 |
| UAEMex #4 | - | - | - | - | - | - | - | - |
| UAEMex #5 | 6.2 | 0.3 | 8.5 | 0.3 | 17.4 | 1.9 | 14.1 | 0.6 |
| UAEMex #6 | 3.4 | 0.0 | 4.7 | 0.0 | 16.1 | 0.0 | 11.2 | 0.0 |
| UAEMex #7 | 0.0 | 0.0 | 0.6 | 0.0 | 1.0 | 0.0 | 1.3 | 0.0 |
| UAEMex #8 | 3.3 | 0.0 | 4.6 | 0.0 | 16.1 | 0.0 | 11.1 | 0.0 |

**Table 4:** Results (in %) for the segment-based performance measures, only for the broken word retrieval results.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | - | - | - | - | - | - | - | - |
| CITlab #1 | 60.1 | 20.2 | 48.6 | 23.8 | 76.4 | 35.5 | 49.8 | 24.5 |
| CITlab #2 | 57.4 | 21.8 | 47.3 | 22.9 | 74.3 | 35.1 | 48.4 | 23.5 |
| CITlab #3 | 59.2 | 20.9 | 47.3 | 22.8 | 75.7 | 33.8 | 48.5 | 23.4 |
| CITlab #4 | 58.4 | 22.1 | 47.0 | 23.0 | 74.2 | 35.5 | 48.2 | 23.7 |
| CITlab #5 | 60.9 | 16.7 | 49.0 | 22.7 | 75.1 | 33.5 | 50.1 | 23.4 |
| CITlab #6 | 60.6 | 19.6 | 48.3 | 22.9 | 76.5 | 34.6 | 49.4 | 23.5 |
| CITlab #7 | 59.4 | 23.6 | 47.9 | 22.8 | 76.0 | 35.8 | 49.0 | 23.5 |
| CITlab #8 | 59.2 | 21.6 | 48.0 | 23.7 | 75.9 | 34.8 | 49.1 | 24.4 |
| CITlab #9 | 60.5 | 17.0 | 49.6 | 23.6 | 75.0 | 34.3 | 50.7 | 24.4 |
| **CITlab #10** | **59.4** | **24.3** | **48.4** | **23.7** | **76.0** | **36.8** | **49.5** | **24.4** |
| IIIT #1 | - | - | - | - | - | - | - | - |
| IIIT #2 | - | - | - | - | - | - | - | - |
| MayoBMI #1 | - | - | - | - | - | - | - | - |
| UAEMex #1 | - | - | - | - | - | - | - | - |
| UAEMex #2 | - | - | - | - | - | - | - | - |
| UAEMex #3 | - | - | - | - | - | - | - | - |
| UAEMex #4 | - | - | - | - | - | - | - | - |
| UAEMex #5 | - | - | - | - | - | - | - | - |
| UAEMex #6 | - | - | - | - | - | - | - | - |
| UAEMex #7 | - | - | - | - | - | - | - | - |
| UAEMex #8 | - | - | - | - | - | - | - | - |

**Table 5:** Results (in %) for the box-based performance measures only for the broken word retrieval results.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | - | - | - | - | - | - | - | - |
| **CITlab #1** | **31.1** | **9.9** | **38.8** | **16.8** | **50.2** | **20.2** | **42.2** | **19.0** |
| CITlab #2 | 26.3 | 9.3 | 35.8 | 15.7 | 45.5 | 18.3 | 39.9 | 17.9 |
| CITlab #3 | 26.1 | 9.5 | 35.8 | 15.6 | 45.8 | 18.1 | 39.9 | 17.8 |
| CITlab #4 | 25.6 | 10.7 | 35.5 | 15.9 | 45.1 | 19.2 | 39.5 | 18.0 |
| CITlab #5 | 31.9 | 7.2 | 38.6 | 16.2 | 49.5 | 18.1 | 42.0 | 18.3 |
| CITlab #6 | 31.4 | 8.9 | 38.0 | 16.2 | 49.8 | 18.8 | 41.4 | 18.3 |
| CITlab #7 | 27.7 | 10.6 | 36.4 | 15.7 | 46.6 | 19.2 | 40.4 | 17.9 |
| CITlab #8 | 26.1 | 10.5 | 36.1 | 16.3 | 46.0 | 19.8 | 40.4 | 18.6 |
| CITlab #9 | 32.0 | 8.0 | 39.7 | 16.9 | 49.9 | 19.5 | 43.1 | 19.0 |
| CITlab #10 | 28.0 | 12.0 | 36.8 | 16.5 | 46.8 | 20.9 | 40.8 | 18.8 |
| IIIT #1 | - | - | - | - | - | - | - | - |
| IIIT #2 | - | - | - | - | - | - | - | - |
| MayoBMI #1 | - | - | - | - | - | - | - | - |
| UAEMex #1 | - | - | - | - | - | - | - | - |
| UAEMex #2 | - | - | - | - | - | - | - | - |
| UAEMex #3 | - | - | - | - | - | - | - | - |
| UAEMex #4 | - | - | - | - | - | - | - | - |
| UAEMex #5 | - | - | - | - | - | - | - | - |
| UAEMex #6 | - | - | - | - | - | - | - | - |
| UAEMex #7 | - | - | - | - | - | - | - | - |
| UAEMex #8 | - | - | - | - | - | - | - | - |

**Table 6:** Results (in %) for the segment-based performance measures, only for the queries with OOV words.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | - | - | - | - | - | - | - | - |
| CITlab #1 | 89.7 | 38.6 | 88.4 | 39.3 | 92.5 | 55.8 | 89.2 | 40.9 |
| CITlab #2 | 87.9 | 32.0 | 88.0 | 37.4 | 92.0 | 53.9 | 89.0 | 39.2 |
| CITlab #3 | 82.5 | 21.7 | 87.0 | 34.9 | 91.2 | 49.2 | 88.1 | 37.2 |
| CITlab #4 | 83.0 | 30.7 | 87.4 | 37.0 | 91.3 | 52.5 | 88.4 | 39.0 |
| CITlab #5 | 88.6 | 28.5 | 88.4 | 38.2 | 92.1 | 52.7 | 89.2 | 39.9 |
| CITlab #6 | 89.3 | 38.0 | 87.8 | 38.7 | 92.2 | 55.3 | 88.7 | 40.4 |
| CITlab #7 | 89.0 | 41.9 | 88.3 | 38.9 | 92.2 | 56.2 | 89.0 | 40.5 |
| CITlab #8 | 82.9 | 21.2 | 87.9 | 35.3 | 91.6 | 49.3 | 88.9 | 37.5 |
| CITlab #9 | 89.0 | 28.8 | 88.7 | 38.9 | 92.4 | 53.1 | 89.5 | 40.5 |
| **CITlab #10** | **89.3** | **42.6** | **88.9** | **39.5** | **92.5** | **56.7** | **89.5** | **41.1** |
| IIIT #1 | 13.2 | 1.3 | 17.6 | 2.8 | 23.0 | 5.6 | 19.9 | 2.9 |
| **IIIT #2** | **13.2** | **1.7** | **17.6** | **2.9** | **23.0** | **6.2** | **19.9** | **3.0** |
| MayoBMI #1 | - | - | - | - | - | - | - | - |
| UAEMex #1 | - | - | - | - | - | - | - | - |
| **UAEMex #2** | **0.2** | **0.0** | **0.9** | **0.2** | **1.7** | **0.2** | **1.1** | **0.2** |
| UAEMex #3 | 0.0 | 0.0 | 0.7 | 0.0 | 1.2 | 0.0 | 1.0 | 0.0 |
| UAEMex #4 | 0.0 | 0.0 | 0.6 | 0.0 | 0.7 | 0.0 | 0.9 | 0.0 |
| UAEMex #5 | 0.1 | 0.0 | 0.5 | 0.0 | 1.3 | 0.0 | 0.7 | 0.0 |
| UAEMex #6 | 0.1 | 0.0 | 0.8 | 0.0 | 1.4 | 0.0 | 1.1 | 0.0 |
| UAEMex #7 | 0.0 | 0.0 | 0.7 | 0.0 | 1.2 | 0.0 | 1.0 | 0.0 |
| UAEMex #8 | 0.1 | 0.0 | 1.0 | 0.0 | 1.4 | 0.0 | 1.3 | 0.0 |

**Table 7:** Results (in %) for the box-based performance measures only for the queries with OOV words.

| **System** | **gAP** | | **mAP** | | **gNDCG** | | **mNDCG** | |
|---|---|---|---|---|---|---|---|---|
| | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** | **Dev.** | **Test** |
| Baseline | - | - | - | - | - | - | - | - |
| **CITlab #1** | **65.6** | **25.0** | **68.4** | **28.4** | **70.7** | **35.4** | **73.3** | **32.0** |
| CITlab #2 | 61.3 | 19.5 | 67.5 | 25.9 | 69.4 | 32.6 | 72.7 | 29.6 |
| CITlab #3 | 57.9 | 15.3 | 66.8 | 24.1 | 68.8 | 31.0 | 72.0 | 28.0 |
| CITlab #4 | 60.4 | 18.3 | 66.9 | 25.6 | 69.3 | 32.0 | 72.2 | 29.4 |
| CITlab #5 | 64.8 | 19.5 | 67.9 | 27.3 | 70.5 | 32.8 | 72.8 | 30.9 |
| CITlab #6 | 65.0 | 23.8 | 67.6 | 27.8 | 70.6 | 34.2 | 72.5 | 31.3 |
| CITlab #7 | 63.4 | 23.9 | 67.8 | 27.4 | 69.9 | 34.0 | 72.7 | 31.0 |
| CITlab #8 | 58.0 | 16.0 | 67.6 | 24.8 | 68.6 | 32.4 | 72.8 | 28.7 |
| CITlab #9 | 65.6 | 20.3 | 68.6 | 27.9 | 70.7 | 33.9 | 73.4 | 31.5 |
| CITlab #10 | 63.6 | 26.0 | 68.4 | 28.4 | 69.8 | 35.8 | 73.4 | 32.1 |
| IIIT #1 | 7.5 | 0.0 | 15.8 | 0.0 | 12.5 | 0.0 | 17.1 | 0.0 |
| **IIIT #2** | **7.5** | **0.0** | **15.8** | **0.0** | **12.5** | **0.0** | **17.1** | **0.1** |
| MayoBMI #1 | - | - | - | - | - | - | - | - |
| UAEMex #1 | - | - | - | - | - | - | - | - |
| **UAEMex #2** | **0.0** | **0.0** | **0.1** | **0.1** | **0.5** | **0.1** | **0.3** | **0.1** |
| UAEMex #3 | 0.0 | 0.0 | 0.1 | 0.0 | 0.4 | 0.0 | 0.3 | 0.0 |
| UAEMex #4 | 0.0 | 0.0 | 0.1 | 0.0 | 0.2 | 0.0 | 0.3 | 0.0 |
| UAEMex #5 | 0.0 | 0.0 | 0.1 | 0.0 | 0.4 | 0.0 | 0.2 | 0.0 |
| UAEMex #6 | 0.0 | 0.0 | 0.1 | 0.0 | 0.4 | 0.0 | 0.3 | 0.0 |
| UAEMex #7 | 0.0 | 0.0 | 0.1 | 0.0 | 0.4 | 0.0 | 0.3 | 0.0 |
| UAEMex #8 | 0.0 | 0.0 | 0.2 | 0.0 | 0.5 | 0.0 | 0.4 | 0.0 |

in which little work has been done. Nevertheless, the CITlab performance for the broken words is quite good, which makes it the most interesting outcome of this evaluation.

## 5 Post-evaluation Dataset Usage

As observed in subsection 4.2, the selected test set was significantly different to the training and development sets, something that was a surprise for both the participants and us the organizers. Among the differences observed are: scanning and image quality, providing only the baselines, new writers or style and differing paper formats. At the moment it is not known exactly what are the differences that are affecting most the system performance, so a more in-depth analysis should be conducted to understand it.

In future usage of this test set, most of the improvements in performance are expected to be due to better handling of these data differences, not the specific challenges proposed in this evaluation. So we are recommending not using it for comparative future works. Since the development set is relatively large and its retrieval ground truth is now publicly available, for subsequent works it is recommended to use only the development set for measuring the performance of the systems. The evaluation protocol and performance measures should be the same as the one proposed in this paper. So that future results are comparable with the ones presented in this paper, a list of 60 pages selected from the development set is being provided, so that these are used as validation for adjusting parameters and hyper-parameters. This list is exactly the same as the one used by the CITlab group in this evaluation.

## 6 Conclusions

This paper presented an overview of the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task, the first edition of a challenge aimed at developing retrieval systems for handwritten documents. Several novelties were introduced in comparison to other recent related evaluations, specifically: multiple word queries, finding local blocks of text, results in transitions between consecutive pages, handling words broken between two lines, words unseen in training and queries with zero relevant results. The evaluation was organized so that groups from several research communities could participate: handwritten text recognition, query-by-string and query-by-example keyword spotting and information retrieval.

The interest in the task was considerable, 48 groups registered, and 24 signed an agreement to get access to the data. In the end, the participation was low, submissions were received for only four groups. The results presented in this paper are for 21 valid system runs received, left after removing 7 submissions that were invalid and 9 that were withdrawn since part of the development set was used for training, something that was prohibited by the rules of participation.

The best performance was obtained by the CITlab. Their system used multidimensional recurrent neural networks (MDRNN) trained using connectionist temporal classification (CTC), followed by regular expression based decoder aided by a word uni-gram language model. Very good performance was obtained for words not observed in the training data and for words broken between lines. Thus, even though the low participation prevented a much more fruitful comparison of systems, the results ended up being very interesting.

This evaluation should serve to give push to this area of research, in particular the novel challenges proposed. The two groups that addressed the broken words problem, based it on locating the hyphenation symbols that generally mark them. However, not always broken words are marked with a special symbol, so future work could be targeted at handling this more general case. Possibly there will be a need to produce new datasets to better assess the challenges proposed here. Not only to have more data with broken word examples, but also to measure performance with queries written in a more natural language, including words that may or may not appear in a relevant document segment, or for example appearing as synonyms.

## Acknowledgments

This work has been partially supported by the European Union (EU) Horizon 2020 grant READ (Recognition and Enrichment of Archival Documents) (Ref: 674943), EU project HIMANIS (JPICH programme, Spanish grant Ref: PCIN-2015-068), MINECO/FEDER, UE under project TIN2015-70924-C2-1-R and the Spanish MEC under FPU grant FPU13/06281.

## References

1. Aldavert, D., Rusinol, M., Toledo, R., Llados, J.: Integrating Visual and Textual Cues for Query-by-String Word Spotting. In: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on. pp. 511–515 (Aug 2013), doi:10.1109/ICDAR.2013.108
2. Causer, T., Terras, M.: 'many hands make light work. many hands together make merry work': Transcribe bentham and crowdsourcing manuscript collections. Crowdsourcing Our Cultural Heritage pp. 57–88 (2014)
3. Causer, T., Wallace, V.: Building a volunteer community: results and findings from Transcribe Bentham. Digital Humanities Quarterly 6 (2012), http://www.digitalhumanities.org/dhq/vol/6/2/000125/000125.html
4. Fischer, A., Keller, A., Frinken, V., Bunke, H.: Lexicon-free handwritten word spotting using character HMMs. Pattern Recognition Letters 33(7), 934 – 942 (2012), Special Issue on Awards from ICPR 2010. doi:10.1016/j.patrec.2011.09.009
5. Frinken, V., Fischer, A., Bunke, H.: A Novel Word Spotting Algorithm Using Bidirectional Long Short-Term Memory Neural Networks. In: Schwenker, F., El Gayar, N. (eds.) Artificial Neural Networks in Pattern Recognition, Lecture Notes in Computer Science, vol. 5998, pp. 185–196. Springer Berlin / Heidelberg (2010), doi:10.1007/978-3-642-12159-3_17

6. Garcá Calderón, M.A., Garcá Hernández, R.A.: UAEMex at ImageCLEF 2016: Handwritten Retrieval. In: CLEF2016 Working Notes. CEUR Workshop Proceedings, vol. 1609. CEUR-WS.org, Évora, Portugal (September 5-8 2016)
7. Gatos, B., Pratikakis, I.: Segmentation-free word spotting in historical printed documents. In: Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on. pp. 271–275 (July 2009)
8. Giotis, A., Gerogiannis, D., Nikou, C.: Word Spotting in Handwritten Text Using Contour-Based Models. In: Frontiers in Handwriting Recog. (ICFHR), 2014 14th Int. Conf. on. pp. 399–404 (Sept 2014), doi:10.1109/ICFHR.2014.73
9. Kozielski, M., Forster, J., Ney, H.: Moment-based image normalization for handwritten text recognition. In: Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on. pp. 256–261 (Sept 2012), doi:10.1109/ICFHR.2012.236
10. Liu, S., Wang, Y., Mehrabi, S., Li, D., Liu, H.: MayoBMI at ImageCLEF 2016 Handwritten Document Retrieval Challenge. In: CLEF2016 Working Notes. CEUR Workshop Proceedings, vol. 1609. CEUR-WS.org, Évora, Portugal (September 5-8 2016)
11. Pratikakis, I., Zagoris, K., Gatos, B., Louloudis, G., Stamatopoulos, N.: ICFHR 2014 Competition on Handwritten Keyword Spotting (H-KWS 2014). In: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on. pp. 814–819 (Sept 2014), doi:10.1109/ICFHR.2014.142
12. Puigcerver, J., Toselli, A.H., Vidal, E.: Icdar2015 competition on keyword spotting for handwritten documents. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 1176–1180 (Aug 2015), doi:10.1109/ICDAR.2015.7333946
13. Puigcerver, J., Toselli, A.H., Vidal, E.: Probabilistic interpretation and improvements to the hmm-filler for handwritten keyword spotting. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 731–735 (Aug 2015), doi:10.1109/ICDAR.2015.7333858
14. Robertson, S.: A new interpretation of average precision. In: Proc. of the International ACM SIGIR conference on Research and development in information retrieval (SIGIR '08). pp. 689–690. ACM, New York, NY, USA (2008), doi:10.1145/1390334.1390453
15. Rodríguez-Serrano, J.A., Perronnin, F.: Handwritten word-spotting using hidden Markov models and universal vocabularies. Pattern Recognition 42, 2106–2116 (September 2009), doi:10.1016/j.patcog.2009.02.005
16. Strauß, T., Grüning, T., Leifert, G., Labahn, R.: CITlab ARGUS for Keyword Search in Historical Handwritten Documents - Description of CITlab's System for the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task. In: CLEF2016 Working Notes. CEUR Workshop Proceedings, vol. 1609. CEUR-WS.org, Évora, Portugal (September 5-8 2016)
17. Sánchez, J.A., Toselli, A.H., Romero, V., Vidal, E.: Icdar 2015 competition htrts: Handwritten text recognition on the transcriptorium dataset. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 1166–1170 (Aug 2015), doi:10.1109/ICDAR.2015.7333944
18. Toselli, A.H., Juan, A., Keysers, D., González, J., Salvador, I., Ney, H., Vidal, E., Casacuberta, F.: Integrated Handwriting Recognition and Interpretation using Finite-State Models. Int. Journal of Pattern Recognition and Artificial Intelligence 18(4), 519–539 (June 2004), doi:10.1142/S0218001404003344

19. Toselli, A.H., Puigcerver, J., Vidal, E.: Context-aware lattice based filler approach for key word spotting in handwritten documents. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 736–740 (Aug 2015), doi:10.1109/ICDAR.2015.7333859
20. Villegas, M., Müller, H., García Seco de Herrera, A., Schaer, R., Bromuri, S., Gilbert, A., Piras, L., Wang, J., Yan, F., Ramisa, A., Dellandrea, E., Gaizauskas, R., Mikolajczyk, K., Puigcerver, J., Toselli, A.H., Sánchez, J.A., Vidal, E.: General Overview of ImageCLEF at the CLEF 2016 Labs. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Lecture Notes in Computer Science, Springer International Publishing (2016)
21. Villegas, M., Puigcerver, J., Toselli, A.H.: ImageCLEF 2016 Bentham Handwritten Retrieval Dataset (2016), doi:10.5281/zenodo.52994
22. Villegas, M., Romero, V., Sánchez, J.A.: On the Modification of Binarization Algorithms to Retain Grayscale Information for Handwritten Text Recognition. In: 7th Iberian Conference on Pattern Recognition and Image Analysis, LNCS, vol. 9117, pp. 208–215. Springer, Santiago de Compostela (Spain) (Jun 2015), doi:10.1007/978-3-319-19390-8_24
23. Zagoris, K., Ergina, K., Papamarkos, N.: Image retrieval systems based on compact shape descriptor and relevance feedback information. Journal of Visual Communication and Image Representation 22(5), 378 – 390 (2011)