International Journal on Document Analysis and Recognition
DOI: 10.1007/s10032-009-0083-y

A Performance Evaluation Protocol for Symbol Spotting Systems in Terms of Recognition and Location Indices

Marçal Rusiñol · Josep Lladós

Abstract Symbol spotting systems are intended to retrieve regions of interest from a document image database where the queried symbol is likely to be found. They shall have the ability to recognize and locate graphical symbols in a single step. In this paper we present a set of measures to evaluate the performance of a symbol spotting system in terms of recognition abilities, location accuracy and scalability. We show that the proposed measures allow to determine the weaknesses and strengths of different methods. In particular we have tested a symbol spotting method based on a set of four different off-the-shelf shape descriptors.

Keywords Performance Evaluation, Symbol Spotting, Graphics Recognition.

1 Introduction

Performance evaluation methods are essential tools to understand and compare the behavior of algorithms and systems. A performance evaluation protocol should identify the strengths and weaknesses of the methods under test. The analysis of these strong points and drawbacks should determine which method is the most suitable for a certain use case and predict its behavior when using it in real applications with real data.

In the last years, performance evaluation has been a quite prolific research topic in the Document Image Analysis and Recognition (*DIAR*) field. Several competitions focused on particular topics, namely, symbol recognition, layout analysis, text detection among others, have been organized in the major conferences and workshops of this field. We can also find a lot of contributions in the recent literature proposing evaluation techniques for different *DIAR* applications. Performance evaluation does not only focus on feature level techniques such as line and arc detection algorithms [19, 20] or raster-to-vector systems [34], but is also useful to evaluate higher level applications such as symbol recognition [40] or layout analysis [2].

In this paper we propose a set of measures and methodologies to evaluate the performance of spotting systems. Although we mainly focus on the specific case of symbol spotting, these measures are also applicable to performance evaluation of other applications such as word spotting, or even object detection in Computer Vision applications. Symbol spotting was first introduced by Tombre and Lamiroy in [37] and can be defined as a way to efficiently locate graphical symbols in document images without using full recognition methods. Such systems are intended to index large collections of document images in terms of the graphical symbols which appear in them. Given a graphical symbol as query, the system has to retrieve a ranked list of locations where the query symbol is likely to be found. Since spotting systems deal with recognition and segmentation at the same time, such abilities must be taken into account by the evaluation process. Segmentation errors must be punished as well as recognition mistakes.

As we illustrate in the review provided in section 2, there exist many approaches to measure the performance of different Graphics Recognition algorithms. However, in the particular case of symbol spotting, existing methods in the literature just provide measures based on binary decisions of *found / not found*. We develop the theory in this paper that the performance of a symbol spotting system should be defined in terms of two components: the recognition and the location goodness. Starting from this hypothesis, the main con-

M. Rusiñol · J. Lladós
Computer Vision Center, Dept. Ciències de la Computació
Edifici O, Universitat Autònoma de Barcelona
08193 Bellaterra (Barcelona), Spain
E-mail: {marcal,josep}@cvc.uab.es

tribution of this paper is to propose a set of performance evaluation measures, based on the precision and recall concepts, to evaluate the performance of symbol spotting systems in terms of two criteria, namely recognition and location. In addition, a second contribution is to use the same formalism to evaluate a third quality criterion, the scalability under an increasing number of symbol prototypes. Most of the work found in the literature dealing with performance evaluation of Graphics Recognition systems is mainly focused on the computation of a score to allow an easy way to rank different methods. We strongly believe that the proposed measures can give a more accurate idea of the real behavior of the system under study than typical recognition rates.

The remainder of this paper is organized as follows: We briefly overview in section 2 the work on performance evaluation for related areas such as retrieval systems, Graphics Recognition and Document Image Analysis applications. In section 3, we basically review the well-known measures of precision and recall typically used in retrieval evaluation and the measures we can derive from precision and recall. Section 4 outlines how these measures can be reformulated and applied to evaluate a spotting system in terms of retrieving regions of interest from a document image database. Section 5 shows a use case of such measures, evaluating the performance of a symbol spotting method based on a set of four different off-the-shelf shape descriptors. Finally, the conclusions and a short discussion can be found in section 6.

2 Related Work

Symbol spotting systems are intended to produce a ranked list of regions of interest cropped from the document images stored in the database where the queried symbol is likely to be found. Symbol spotting can thus be seen as a particular application within the Information Retrieval (*IR*) domain. Usually, retrieval systems are evaluated by *precision* and *recall* ratios which give an idea about the relevance and the completeness of the results (we will briefly review these measures in section 3). These basic measures can be enhanced considering many other indicators depending on the application. For instance, Lu et al. evaluate in [22] a set of desktop search engines by deriving a set of ratios from precision and recall to indicate the abilities of the systems when incrementally retrieving documents. Müller et al. evaluate in [27] content-based image retrieval systems, proposing some strategies to take into account the way the number of items stored in the collection affects the results and how user feedback can improve the response of such systems. Kang et al. evaluate in [16] a text retrieval system which uses semantic indexing, focusing on the distribution and amount of key-indices used to index the database. Finally, we can find in [12, 28] the performance analysis of some *IR* systems having the information distributed in a peer-to-peer network (*P2PIR*), which takes into account the query response time, the network resources requirements and the tradeoff between distributed and centralized systems. As we can see, the coverage of *IR* topic is so wide that even if researchers use similar indicators to evaluate the performance of their methods, no general evaluation framework can be defined. In our case we will also base our measures on the notions of precision and recall by adapting them to the recognition and location abilities that the spotting systems should present.

In the Document Image Analysis and more particularly the Graphics Recognition field, some work focused on spotting can be found. However all this work is evaluated by ad-hoc measures. For instance, Rath and Manmatha presented in [30] a system able to spot handwritten words in ancient documents. They evaluate their system with a score based only on the precision value. Marcus presented in [26] an algorithm to spot spoken words in an audio signal. The evaluation is based on Receiver Operating Characteristics (*ROC*) graphs [11] which are related to precision and recall measures. Tabbone and Zuwala present in [36] a method to spot graphical symbols in a collection of electronic drawings. They base the evaluation of their method in precision and recall graphs. Finally, Valveny et al. present in [40] a framework to evaluate symbol recognition methods envisaging a way to evaluate location and recognition of symbols by also using precision and recall measures. However, all these methods are computed on a binary retrieval notion: whether an item is considered retrieved or not. By these measures one can see the ability of the system in retrieving relevant items and discarding negative ones, but these measures do not evaluate how well the system located the queried objects.

To avoid binary relevance labelling, our measurements are inspired in the techniques used to evaluate layout analysis systems. In fact, layout analysis shares some similarities with spotting in the sense that subregions from documents have to be labelled according to their content. Layout analysis competitions [3–5] were held in last editions of the *ICDAR* conference. In these contests, the evaluation of the participants' methods was done according to the overlapping between regions of the results and the ground-truth. Two indicators introduced in [29] are used to formulate an entity detection measure from which an averaged segmentation measure is deducted to score the systems.

Following the same idea, in the text detection competitions [23,24] held in last editions of *ICDAR*, precision and recall measures were computed in terms of overlapping between bounding-boxes of the ground-truth and the results. From the precision and recall numbers, a score was computed to rank the algorithm performance. However we believe that the use of a single evaluation score allow an easy ranking of the different systems, but hinders the understandability of their behavior and the performance prediction when using other type of datasets.

Finally, in the last symbol recognition competitions [1,38,39] held in the last *GREC* workshop editions, several symbol descriptors where evaluated. In that case, the performance is evaluated by the recognition rates the systems yield. In the last edition, other measures such as the homogeneity and the separability of the symbol classes in the description space have been introduced. We find very interesting the fact that the scalability of the systems is also tested. This test is performed looking how the performance of the systems evolve as the number of symbol classes to consider increases.

The measures we propose in this paper are based on precision and recall, since it has been demonstrated to be a good way to evaluate recognition (or at least classification) and location at the same time. We formulate the precision and recall notions in terms of overlapping between retrieved areas and ground-truth. The presented measures and plots allow to assess the weaknesses and strengths of the methods in terms of recognition abilities and location accuracy. In addition we also present a methodology to extract a scalability measure from precision and recall to test if the methods can be used with a larger amount of classes. Let us first review the basic measures used to evaluate retrieval effectiveness.

3 An Overview on Measures to Evaluate Retrieval Effectiveness

In this section we review the basic measures provided in the literature used to evaluate the retrieval effectiveness. The measures outlined in this section will be reformulated in section 4 for the framework described in this work.

3.1 Precision and Recall

In the *IR* field, most measures to evaluate effectiveness are based on a binary labelling of relevance of the items, namely whether each item is considered as relevant or non-relevant. In addition, these measures are also based on a binary retrieval notion, i.e. whether an item is retrieved or not.

Given a database consisting of a set of elements tot, and a query item i to retrieve from it, let us label as rel the set of relevant objects in the set and $\overline{rel}$ the set non-relevant items with regard to the query i. When querying this item to the database, we label as ret the set of retrieved elements and as $\overline{ret}$ the set of elements from the database which were not retrieved. The retrieval matrix of Table 1 shows all the possibilities in terms of intersections between these sets.

Table 1: Retrieval Matrix

	Relevant	Non-Relevant	TOTAL
Retrieved	$\|ret \cap rel\|$	$\|ret \cap \overline{rel}\|$	$\|ret\|$
Not Retrieved	$\|\overline{ret} \cap rel\|$	$\|\overline{ret} \cap \overline{rel}\|$	$\|\overline{ret}\|$
TOTAL	$\|rel\|$	$\|\overline{rel}\|$	$\|tot\|$

The analysis of this table allows to define the well-known ratios of precision and recall (see van Rijsbergen's [31] book on Information Retrieval for more details) to evaluate the behavior of the *IR* system which are computed as follows:

$$P = \frac{|ret \cap rel|}{|ret|}, \qquad R = \frac{|ret \cap rel|}{|rel|} \tag{1}$$

For a given retrieval result, the *precision* measure P is defined as the ratio between the number of relevant retrieved items and the number of retrieved items. The precision measure measures the quality of the retrieval system in terms of the ability of the system to only include relevant items in the result. A hundred percent precision means that no false positive has been included in the system response. As the precision value decreases, the more non-relevant items are included in the results.

The *recall* ratio R is defined as the number of relevant retrieved items as a ratio to the total number of relevant items in the collection. It measures the effectiveness of the system in retrieving the relevant items. A hundred percent recall means that all the items labelled as relevant are retrieved and no one has been missed. As the recall value decreases, the more relevant items are missed by the system which wrongly considers them as non-relevant.

3.2 $P@n$ and $P(r)$

The precision and recall measures are computed on the whole set of items returned by the system. That is,

they give information about the final performance of the system after processing a query and do not take into account the quality of ranking in the resulting list. *IR* systems return results ranked by a confidence value. The first retrieved items are the ones the system believes that are more likely to match the query. As the system provides more and more results, the probability to find non-relevant items increases.

Relevance ranking can be evaluated computing the precision at a given cut-off rank, considering only the n topmost results returned by the system. This measure is called precision at n or $P@n$. However, this measure presents the drawback that it does not give information about recall.

Let us define $P(r)$ as the precision at a given recall cut-off, that is the precision at that point where recall has first reached the value r.

3.3 Precision and Recall Plots

The usual way to represent the stability of the system as the user requires more and more results is to plot precision and recall against each other. Such plots are computed stepwise retrieving at each step a given item while varying the decision threshold value over the confidence rate, i.e. computing $P@n$ for the different values of n and plot this values against its associated recall.

These plots show the tradeoff between precision and recall. Buckland and Gey analyzed in [8] the relationship between both ratios concluding that they are inversely related, trying to increase one usually provokes the other to be reduced. Thus, when comparing several methods, the one yielding the higher values for both precision and recall will be the best. However, it is not always easy to assess which precision and recall plot corresponds to a better system.

3.4 Measures of Quality

Sometimes it is difficult to measure the effectiveness by a measure composed by more than a number. The difficulty in certain cases to assess which method is the best, has led to invest in some composite measures which can rank the methods under study according to a combination of precision and recall information. However, as claimed in [31], usually these measures are rather ad-hoc and difficult to interpret.

Let us see a couple of composite measures which try to combine both precision and recall information in a single number.

3.4.1 Average Precision

We can define the average precision $AveP$ using each precision value after truncating at each relevant item in the ranked list resulting after a query. Average precision is one of the evaluation measures used by the *TRECVid*[1] community [33].

For a given query, let $\mathrm{r}(n)$ be a binary function on the relevance of the nth item in the returned ranked list, we define the average precision as follows:

$$AveP = \frac{\sum_{n=1}^{|ret|}(P@n \times \mathrm{r}(n))}{|rel|} \tag{2}$$

The average precision is a measure of quality which rewards the earliest return of relevant items. Retrieving all relevant items in the collection and ranking them perfectly will lead to an average precision of 1. The average precision can also be seen as the area under the precision and recall plot. However, average precision does not take into account the fact that a system returns non-relevant items after having reached a hundred percent recall (i.e. having returned all relevant items).

3.4.2 F-score

Another classical composite measure is the F-score (see [13] for more details) which is the weighted harmonic mean of precision and recall, computed as follows:

$$F^{\beta} = \frac{(1+\beta^2) \times P \times R}{(\beta^2 \times P) + R} \tag{3}$$

Which for a value of $\beta = 1$ is equivalent to Dice's coefficient (a well-known similarity measure between two sets X and Y) defined as:

$$s = \frac{2|X \cap Y|}{|X| + |Y|} \tag{4}$$

Although there is some work like [25] which point out some drawbacks of this measure, the F-score is widely used as a measure of merit in the *IR* literature.

The F-score can also be computed at several recall cut-offs to evaluate the stability of a system's response. We re-formulate the F-score presented in eq. 3 for several recall values as:

$$F^{\beta}(r) = \frac{(1+\beta^2) \times P(r) \times r}{(\beta^2 \times P(r)) + r}, \text{ with } r \in [0, R] \tag{5}$$

We can see some examples on how $F^1(r)$-score evolves in Fig. 1 for several synthetic precision and recall plots. The better the system responds, the higher its values. As we can appreciate, F-score heavily penalizes low values of precision or recall.

[1] TREC Video Retrieval Evaluation (http://www-nlpir.nist.gov/projects/trecvid/)

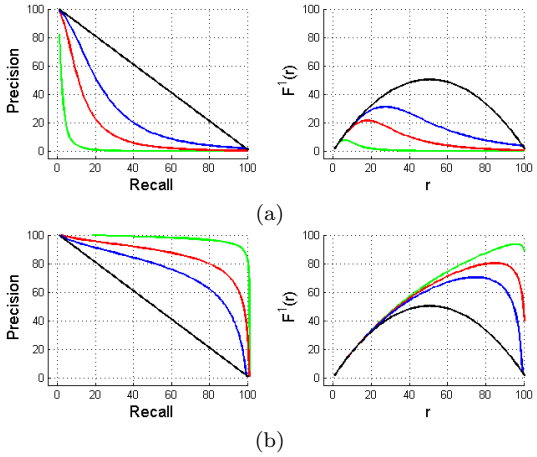


Fig. 1: $F^1(r)$-score plots for different synthetic precision and recall plots.

3.5 Fall-Out and Generality

Let us finally introduce two more measures, one related to the non-relevant retrieved items and the other related to the dataset, which are computed as follows:

$$Fo = \frac{|ret \cap \overline{rel}|}{|\overline{rel}|}, \qquad G = \frac{|rel|}{|tot|} \tag{6}$$

The *fall-out* ratio Fo gives information about the number of non-relevant retrieved items in respect to the number of non-relevant items present in the collection. Independent of the precision of a system, this measure should have low values to consider the behavior of the system good. Either because very few non-relevant items have been retrieved or because the number of non-relevant retrieved items is negligible in relation to the number of non-relevant items in the dataset. To evaluate the evolution of the systems response in terms of false positives usually the fall-out is plotted against recall. This plot is equivalent to the typical *ROC* graphs [11], which are commonly used to evaluate the performance of classifiers. We can find in [9] a study of the relationship between precision-recall and *ROC* curves.

Finally, the *generality* ratio G, gives information about the collection dataset. It is computed as the number of relevant items in the entire collection for a certain query. It can be then averaged for all the considered queries in the experimental setup denoted as the $AveG$ ratio. This ratio does not give any measure about the effectiveness of the retrieval itself, but complements the previous measures. As claimed in [15], when evaluating the performance of a retrieval system, this measure should be given to really understand the meaning of the values of precision, recall and fall-out.

3.6 Central Tendency of Precision and Recall

To evaluate a retrieval system, obviously many queries have to be performed. Each query under evaluation results in a precision and recall plot. To give an idea on well good the system responds, the retrieval results are averaged over these queries. The central tendency of several precision and recall plots are computed sampling individual curves at different points and averaging the samples. We can see an intuitive example of the central tendency in Fig. 2.

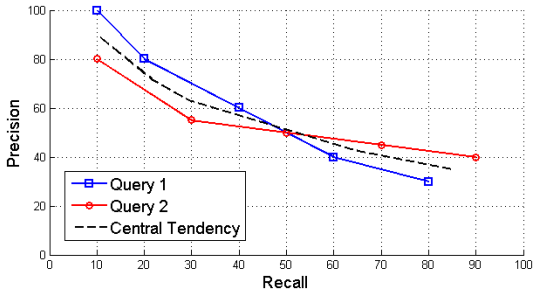


Fig. 2: An example of computing the central tendency of precision and recall plots.

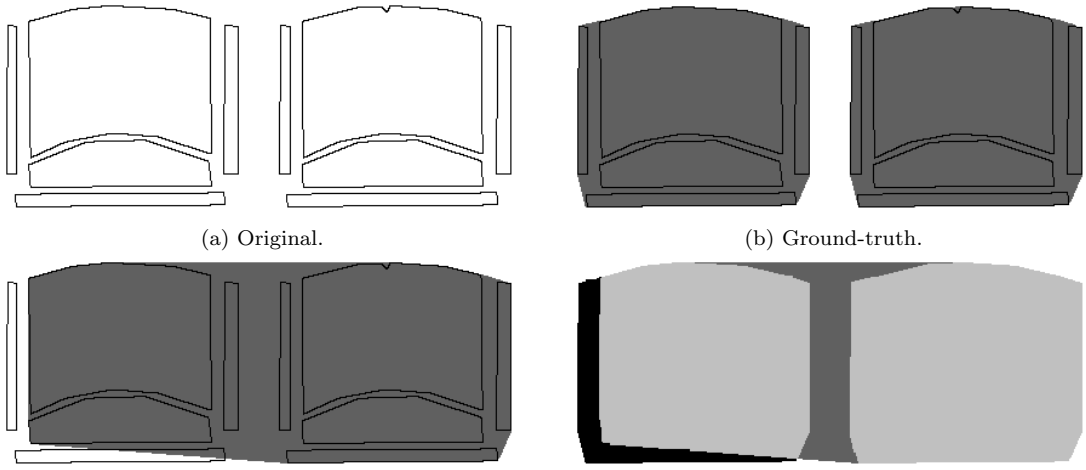
The same averaging technique is applied to fall-out versus recall plots and to $F^\beta(r)$-score plots.

4 Precision and Recall for Spotting Systems

Spotting systems are intended to perform both recognition and location at the same time, and thus, these abilities have to be evaluated together. Let us first propose a formulation of the precision and recall measures to evaluate both concepts. To help the interpretation of precision and recall plots, we propose to use two more measures focused at symbol level which only consider a binary concept of retrieval. Finally, we propose a scalability test to check the systems ability to achieve similar behavior independent of the number of queried symbols.

4.1 Precision and Recall of Regions of Interest

To evaluate the performance of a spotting system we propose a set of measures inspired by both *IR* and layout analysis. The idea is to merge both precision and recall measures with area overlapping rates. Precision and recall ratios provide information on the incremental accuracy of the retrieval process in terms of recognized



(a) Original. (b) Ground-truth.

(c) Result. (d) Polygon Overlapping.

$A(P_i)$	Precision (%)	Recall (%)
$A(\mathrm{P}rel) = 60560$		
$A(\mathrm{P}ret) = 62307$		
$A(\mathrm{P}ret \oplus \mathrm{P}rel) = 55449$	88.99	91.56
$A(\overline{\mathrm{P}ret} \oplus \mathrm{P}rel) = 5111$		
$A(\mathrm{P}ret \oplus \overline{\mathrm{P}rel}) = 6858$		

(e) Polygon Set Areas, Precision and Recall.

Fig. 3: Original image (a), its ground-truth (b) and the result (c) of a spotting system. The overlapping between results and ground-truth (d) is labelled according to $\mathrm{P}ret \oplus \mathrm{P}rel$ (light gray), $\mathrm{P}ret \oplus \overline{\mathrm{P}rel}$ (dark gray) or $\overline{\mathrm{P}ret} \oplus \mathrm{P}rel$ (black). In (e) we can see the detailed areas and obtained precision and recall.

items. On the other hand, the region overlapping between results and ground-truth data is used to evaluate the segmentation accuracy.

To compute the region overlapping between result and ground-truth, we define for both data polygons representing regions of interest. The more accurate is the definition of the region of interest the more the evaluation is reliable. To define the region of interest where a symbol is located we use the convex-hull [7] of all the points belonging to the symbol. In our applications, we usually define the graphical symbols by their external contours, so the convex-hull of the contour pixels englobe the whole symbol. Convex-hulls define much more accurately the zones where a symbol is than bounding-boxes or ellipses. This representation can be extended to different formats of the data of the collection (bitmap or vectorial format) and to different symbol representations (internal pixels, skeleton, contours, segments, etc.).

Given a collection of graphical documents, we denote as Ptot the set of polygons representing the whole document image database. For any graphical symbol S to spot in the collection, we label as Prel the ground-truth polygon set which is composed by all the polygons framing the locations where we find an instance of the symbol S. When spotting S in the document collection, we denote as Pret the set of retrieved polygons. To match the results from the system to the ground-truth polygon set, we define the polygon set intersection operation $\mathrm{P}_k = \mathrm{P}_i \oplus \mathrm{P}_j$, that given two polygon sets P_i and P_j, results in a set of polygons from the spatial overlapping of the polygons belonging to the different sets. To measure this polygon overlapping, we define the function $A(P_i)$ as the sum of areas of all the polygons in the set P_i.

From the above sets and functions, precision and recall ratios of can thus be easily formulated in terms of areas of the overlapping between sets of polygons representing results and ground-truth as follows:

$$P_A = \frac{A(\mathrm{P}ret \oplus \mathrm{P}rel)}{A(\mathrm{P}ret)}, \qquad R_A = \frac{A(\mathrm{P}ret \oplus \mathrm{P}rel)}{A(\mathrm{P}rel)} \tag{7}$$

We can see in Fig. 3 an example of ground-truthed symbols and a result from a spotting system. Some background region has been considered as forming part of the symbol. When we compute the overlapping between retrieved regions and relevant ones this false positive region is identified, resulting in a precision decrease. On the other hand some part of the symbol has been missed, this results to the recall value not reaching one hundred percent.

4.2 Measures of Quality, Fall-out and Generality

Analogously, the measures of quality $AveP$ and F-score, and the ratios fall-out and generality can be expressed in terms of the area of the overlapping between polygon sets representing the ground-truth and the results from the spotting system.

We reformulate eq. 2 by using the area precision at n ($P_A@n$). That is computing the area precision value after truncating the result list after each polygon having some overlapping with a polygon in the ground-truth. The average area precision is then computed as:

$$AveP_A = \frac{\sum_{n=1}^{|\mathrm{P}ret|}(P_A@n \times \mathrm{r}(n))}{|\mathrm{P}rel|} \tag{8}$$

By using the area precision and area recall, we reformulate the F-score from eq. 3 as:

$$F_A^\beta = \frac{(1+\beta^2) \times P_A \times R_A}{(\beta^2 \times P_A) + R_A} \tag{9}$$

and the use of the area precision at a certain area recall cut-off ($P_A(r)$) aim to reformulate eq. 5 as:

$$F_A^\beta(r) = \frac{(1+\beta^2) \times P_A(r) \times r}{(\beta^2 \times P_A(r)) + r}, \text{ with } r \in [0, R_A] \tag{10}$$

Finally, $\overline{\mathrm{P}rel}$ being the complementary polygon set for the ground-truth we can reformulate the fall-out and the generality from eq. 6 as:

$$Fo_A = \frac{A(\mathrm{P}ret \oplus \overline{\mathrm{P}rel})}{A(\overline{\mathrm{P}rel})}, \qquad G_A = \frac{A(\mathrm{P}rel)}{A(\mathrm{P}tot)} \tag{11}$$

4.3 Measures at Symbol Level

As pointed out in [23], sometimes precision and recall based measures are difficult to interpret. A precision of 70% could mean that all symbols were found with an accuracy of 70%, or, on the other hand, that only 70% of the symbols were correctly identified and the other 30% completely missed. A low precision value can be due to a low accuracy in the recognition or to a bad location due to over-segmenting. The recall value can be also affected by missed symbols or by under-segmentation.

To complement the precision and recall based measures, in our experiments we also provide two measures focusing on the recognition at symbol level. In this case we only consider a binary concept of retrieval. Whether a symbol is found or not. Let us consider one symbol S_i and its polygonal representation $\mathrm{P}rel_i$ from the ground-truth, it will be considered as recognized if:

$$A(\mathrm{P}rel_i \oplus \mathrm{P}ret) \geq thr * A(\mathrm{P}rel_i) \tag{12}$$

That is, if the resulting polygons are able to overlap at least a certain percentage of the ground-truthed representation of a symbol, this symbol is considered as recognized. On the other hand, if the resulting polygons do not cover the ground-truth, the symbol is considered as missed. Of course, as with all decisions implying a certain threshold, its value can be critical, and the system's evaluation can depend on it. Its definition is completely subjective as it depends on what the user considers a symbol as being detected or not. The important thing here is that this value is provided when evaluating a system, so as the readers can easily interpret the meaning of the evaluation results. In our case, we consider a symbol as detected if it overlaps at least a 75% with the ground-truth area.

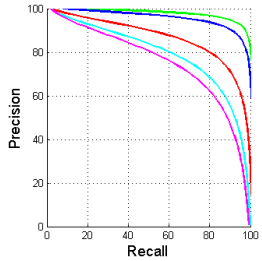
At symbol level, we derive the *recognition rate* of the spotting system under study. In addition, if one of the polygons $\mathrm{P}ret_j$ in the resulting set does not overlap with any recognized symbol, it is considered a *false positive*. For all the possible queries, the average of false positives $AveFP$ is computed. These two measures help to better interpret the values of precision and recall.

Notice that the recognition rate is expressed as a percentage of the total number of symbols in the ground-truth and can be used as a measure of quality by itself, but the false positives are not normalized and are given in absolute values. This is due to the fact that we can not define the negative set in terms of symbol items in the dataset. The false positive average can only be expressed in absolute values and used to compare methods between them.

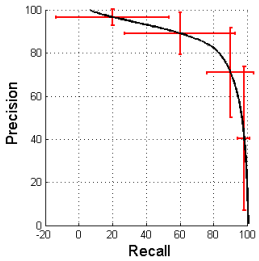
4.4 Scalability Test

Finally, one of the main interests for spotting systems is that a system has to be applicable to a large data corpora. To test the scalability of the system, i.e. its ability to achieve similar behavior independently from the number of queried symbols, we propose a measure to evaluate the scalability of the systems under study.

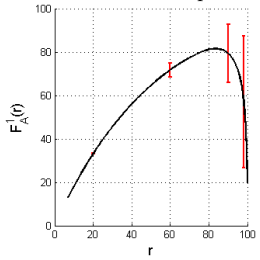
A scalable system has to yield similar responses no matter what the number of model classes taken into account is. We can measure the scalability of a system in terms of its variance in both precision and recall. Let us consider the synthetic example of Fig. 4a which is highly damaged by the addition of new classes. Let us define std_R and std_P the standard deviations in precision and recall for a certain sampling of the precision and recall plot. We can see in Fig. 4b the central tendency of all precision and recall plots with error bars following the vertical and horizontal axis to check the effect in both precision and recall measures when considering more and more classes. The greater the deviation is, the worst the system tolerates changes in the class number, thus the system can be considered as less scalable. To



(a) Synthetic precision and recall plots.



(b) Averaged precision and recall plot with standard deviations in recall and precision.



(c) Averaged $F_A^1(r)$-score plot with associated standard deviations.

Fig. 4: Scalability test example.

allow an easier interpretation, the standard deviation can be computed for the $F_A^\beta(r)$-score plots (as shown in Fig. 4c) having now a single variance measure instead of having one for precision and one for recall. To compare the scalability between different methods, both the mean $\overline{std}$ of all the samples of the standard deviation and the maximum $\max(std)$ of all the samples of the standard deviation are given as variance measures.

On the other hand, the performance of a spotting system not only is affected by the increasing number of considered models but also is dependent on the size of the document collection. To appreciate how the system degrades with the expansion of the dataset we propose to work at symbol level. Recognition rates and false alarms are given to illustrate the performance variability in relation to the size of the database. These measures help to predict how the performance of a system will be affected by the inclusion of more documents in the database. However, increasing the database size has an important drawback. When adding new documents in the database implicitly we can be adding new graphical symbols contained in these new documents. As a consequence of that the number of model symbols has also to be increased along with the dataset size.

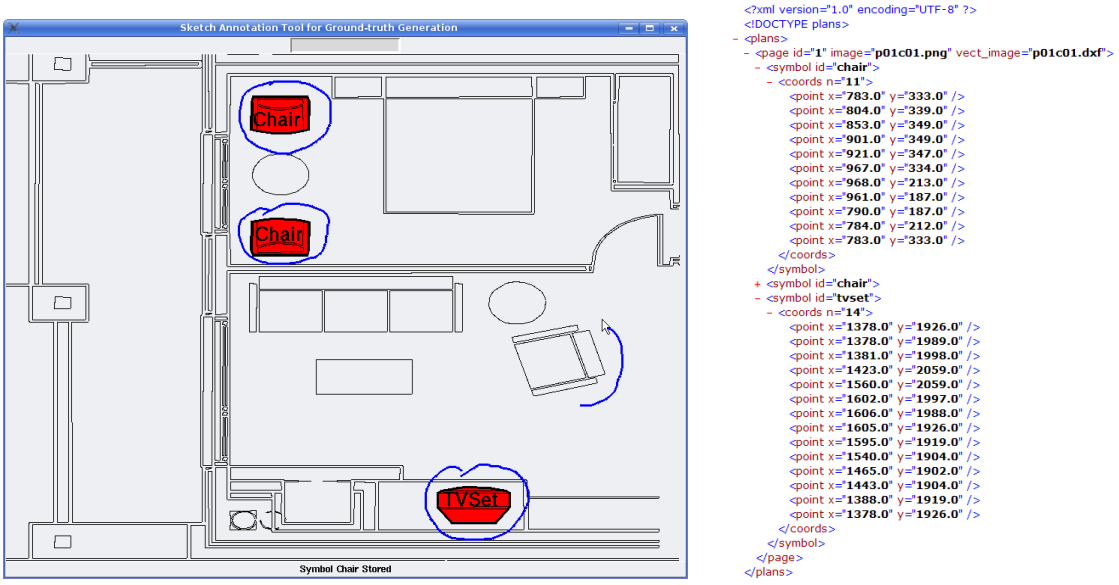
5 Evaluating a Symbol Spotting System

In this section, to show an example of application of the presented evaluation framework, we tested a symbol spotting architecture. We first explain the ground-truthing process, then we briefly detail the spotting system and the used dataset and finally we provide the evaluation results for this architecture. The description of symbol spotting methods is out of the scope of this paper.

5.1 Ground-truthing

First, an annotation tool has been developed to build the ground-truth. The user can select graphical entities in the document images roughly segmenting them using a sketching application. All the contour pixels falling inside the delimited zone of interest are taken as being part of the symbol. If a given connected component has more pixels outside the zone of interest than inside, it is considered as being part of the background. This basic annotation tool works fine with architectural drawings where the symbols are usually not extensively connected with background elements. For other kinds of documents, e.g. electronic diagrams or geographical maps, the annotation tool should be enhanced in order to provide a trusted ground-truth. For all the foreground pixels, we compute the convex-hull as presented in [7] as the minimum area of interest which contains the symbol. Once the region of interest is shown, the user can modify it using certain control points and label them by their content. We can see a screen-shot of the sketching application in Fig. 5a. The use of convex-hulls as the ground-truth primitive may be inadequate for some spotting systems. The inclusion of noisy pixels in the spotting results may provoke considerable deviations of the convex-hull from the one defined in the ground-truth. However, the presented evaluation measures can be easily adapted to other choices of ground-truth primitives. From coarser to more refined primitives we can select for instance to use bounding-boxes, ellipses, isothetic polygons, quad-trees, etc. as ground-truth primitives. In all these cases, the computation of the overlap between ground-truth and automatically extracted primitives is straightforward.

As the user labels the regions containing the graphical symbols, an XML file is constructed to store the information about the whole library. Following the same file structure used for page layout ground-truth presented in [6], the convex-hull coordinates and the symbol category as well as other information about the document are organized in the XML file we can see in Fig. 5b.



(a) (b)

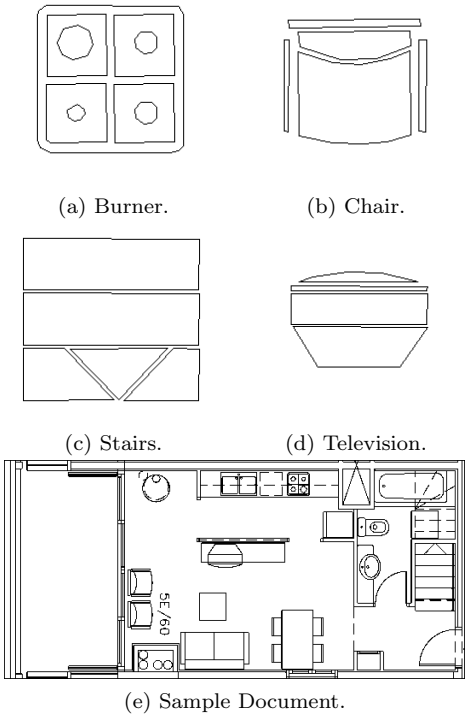
Fig. 5: Sketching annotation tool for ground-truth generation (a) and its ground-truth XML file (b).

As claimed in [21], creating a ground-truth for graphic documents is not always straightforward due to ambiguous cases or subjectivity issues. For example, in the architectural field, each architect tends to use its own symbol designs to represent a furniture element. Whereas human observers have no difficulty in clustering these elements despite the design differences, it is usually impossible for a spotting system to be able to identify different designs as the same object. In the process of ground-truth building, we tried to avoid such problems but we believe that the use of a collaborative framework as proposed in [40] would enhance a lot the quality and the accuracy of this ground-truth. To avoid subjective decisions on the ground-truthing process, synthetic ground-truth can be generated for graphic rich documents, as recently presented in [10]. Such tools which synthetically generate ground-truthed data present several interesting advantages. Subjective decisions are avoided since no human interaction is needed, thus providing an error-free labelling of graphical items. In addition, we have complete control on the number of items in the collection and the number of symbols which have to appear in each document, making the scalability tests much more easy and reliable. However, nowadays the data generated by these methods still appears quite artificial and the use of real data (when possible) should be preferred.

5.2 Spotting Method Under Test

The symbol spotting architecture we use to test the evaluation measures is based on a relational indexing scheme for graphical primitives and a voting methodology which clusters the locations where several hypotheses are casted. These are the locations where it is likely to find the queried symbol. The used spotting system architecture is similar to the one presented in [32]. Symbols are decomposed in basic primitives which are subsequently described by a shape descriptor. The feature vectors arising from the description are indexed by a hashing technique. When querying this hash table, a relational indexing technique is applied. That is, that only similar primitives sharing the same spatial relationship are retrieved. One of the most important points of the system is the way the graphical primitives are described to be indexed. We tested four off-the-shelf shape descriptors described below.

- **Method a**: uses a set of simple ratios described in [35] such as the eccentricity or the non-circularity as shape descriptors. These rough descriptors are formulated from the shape contour of the symbol's primitives. It is expected that the use of such simple shape description can only discriminate very dissimilar shapes; the system should result in a lot of false alarms but should be tolerant to distortions and thus retrieve almost all the instances of the queried symbol.
- **Method b**: uses Hu's geometric invariants [14] to describe contours. These invariants are known as



(a) Burner. (b) Chair.

(c) Stairs. (d) Television.

(e) Sample Document.

Fig. 6: Symbol models and an example of a document in the database.

good shape descriptors. The expected performance is to have good spotting rates in all aspects.

- **Method c**: is based on a reformulation of the previous one. Geometric moments can be formulated for polygonally approximated contours [18] which are taken as primitives. In this case, the use of simpler primitives should result to smaller tolerance to distortions.
- **Method d**: uses the Fourier transform to compactly represent a curvature signature computed over the shape contour. This descriptor is detailed in [17]. This is also a good shape descriptor and the systems performance is expected to be good in all aspects.

Note that we do not want to perform an exhaustive evaluation of shape descriptors or primitives. These methods have been chosen because of their different nature and to test if the proposed evaluation measures really determine the strong and weak points of each method. As the descriptors are well-known among the Graphics Recognition community, it is easy to assess whether the results correspond to the expected behaviors.

5.3 Dataset

The dataset is a collection of architectural floorplans consisting of 42 images (of 3215 × 2064 pixels in average) arising from four different projects. Any given furniture symbol appears in several images in the database. The symbols taken into account for these experiments are divided into 38 classes and we have in total 344 instances in the document images. In a single document image the average of symbols is around 8 ranging from 0 to 28 symbols. The models to query the document database are cropped from the document images. We can see in Fig. 6 some examples of model symbols as well as a sample document from the database.

5.4 Evaluation

We first present the plots showing precision versus recall and fall-out versus recall in Fig. 7 for all the four spotting methods under evaluation using the whole collection of documents. Methods b and d show an acceptable tradeoff between precision and recall as expected. Method d misses much more symbols than method b but gives a significantly smaller amount of false positives. Method a yields good recall values, i.e. it succeeds in retrieving most of the symbols in the document database but has a poor precision due to the high amount of false positives. Finally, method d shows good precision values at early recall stages but quickly falls missing more than half of the symbols in the dataset. The proposed measures aim to stress the expected good behavior of methods b and d and to point out the simplicity of method a and the lack of tolerance of method c.

We can appreciate in Fig. 8a the $F_A^1(r)$-score plots. In this graph we can see again the clear dominance of methods b and d over the other two. As the F-score combines both precision and recall, the methods which fail in one of those measures are clearly demoted in the overall evaluation. Method a starts with a low precision value while the precision of method c quickly falls stopping at a 50% recall. Those two methods are clearly at disadvantage as expected. In Fig. 8b we can see how we can use the $F_A^1(r)$-score plots to visually check the variance of performance of a given method depending on the symbol the user queries.

In Table 2 we can see the measures of quality for all the methods. As the average precision $AveP_A$ measure does not take into account the recall, the method d is ranked as the best. On the other hand, F_A^1-score gives the best for method b. The measures working at symbol level, which are intended to evaluate only the recognition task, are consistent with the results shown in Fig. 7b. The amount of recognized symbols is related to the recall value, which ranks the methods in the order a,b,d and c, in terms of the amount of correctly retrieved elements. On the other hand, the average of false positives is related to the fall-off ratio, ranking the methods

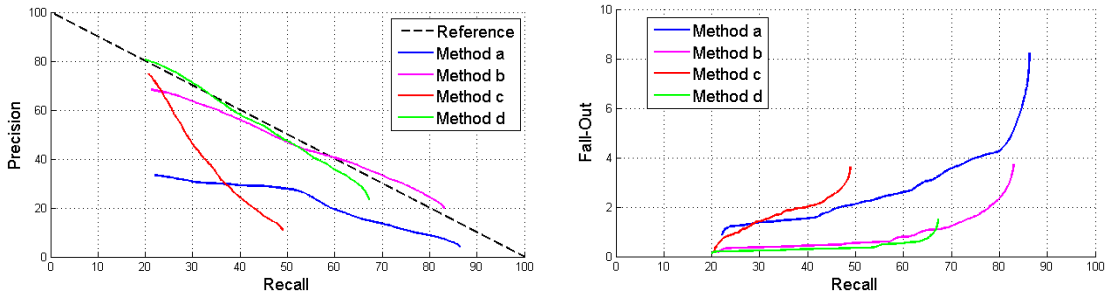


Fig. 7: Precision versus recall is shown in (a) and fall-out versus recall in (b).

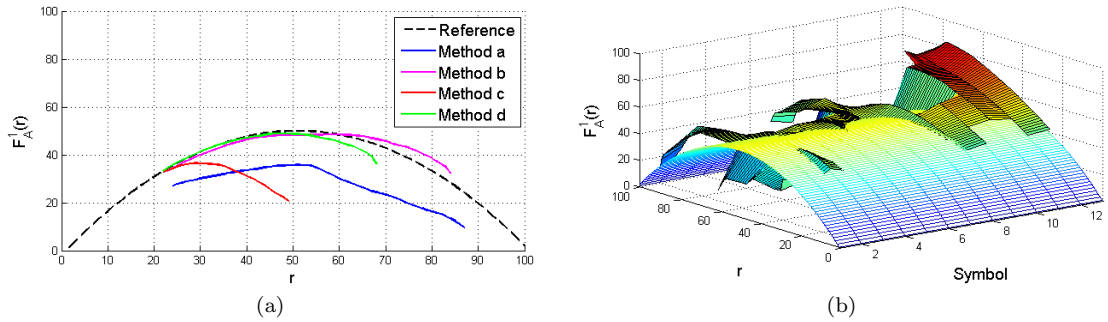


Fig. 8: $F_A^1(r)$-score plot for all methods under test is shown in (a) and $F_A^1(r)$-score plot depending on the queried symbol for method b is shown in (b).

Table 2: Measures of quality.

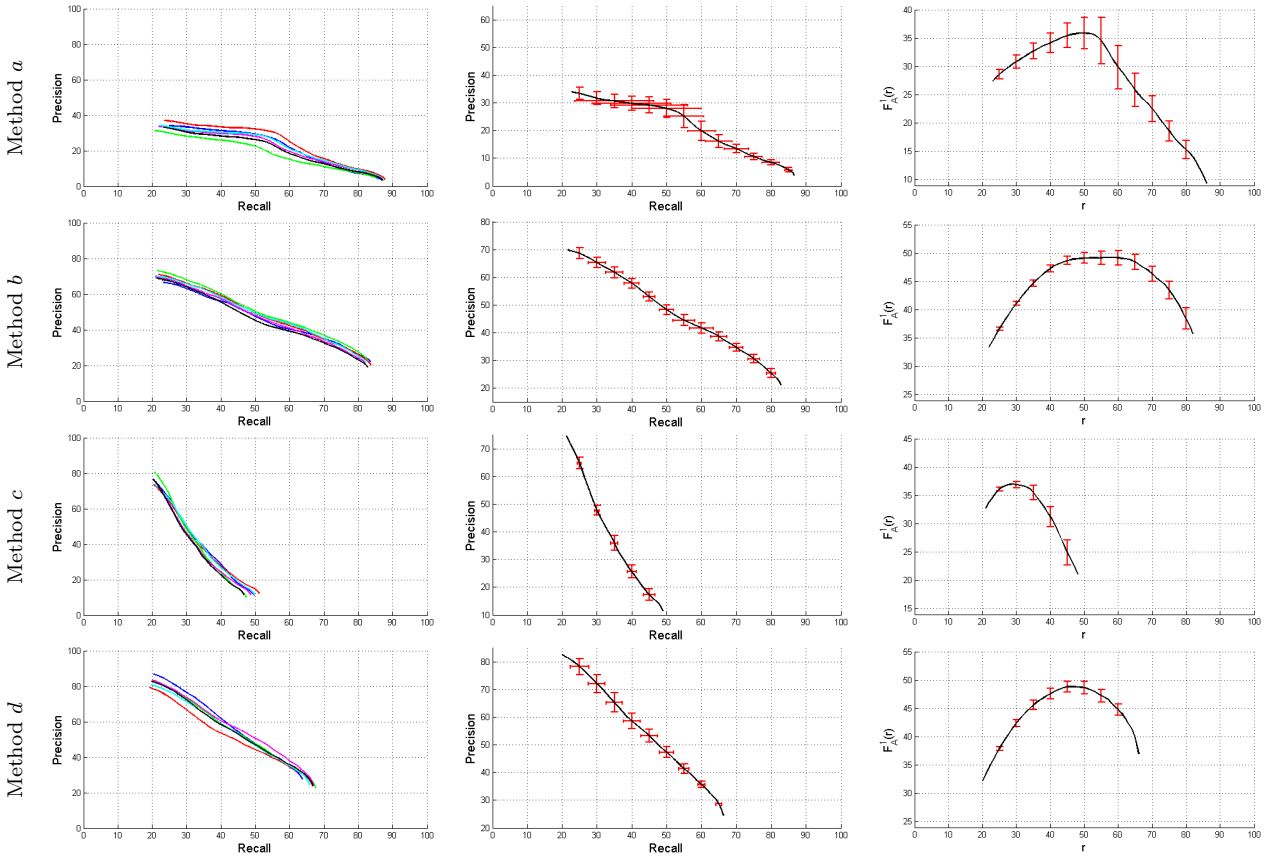
Method	$AveP_A$	F_A^1-score	Rec. rate (%)	$AveFP$	$AveG_A$ (%)
a	20.08	6.87	**93.62**	153.42	0.16
b	39.77	**23.34**	91.3	76.76	
c	23.69	12.57	55.62	63.89	
d	**41.99**	21.45	73.33	**58.76**	

in the order d,c,b and a, in terms of the false alarms present in the results. Finally, the averaged generality gives an idea of the proportion between relevant and total elements in the dataset. These last measures aim to interpret the precision, recall and fall-out values. For spotting applications it is typical to have an extremely low generality measure, since usually the documents in the collections will have much more background objects than foreground ones. This low generality explains the low precision values in both precision and recall plots and in the average precision $AveP_A$ indicator.

Finally, the scalability test results are shown in Fig. 9. Several sets of symbol classes are considered ranging from only 5 to 35 possible symbols to query. We randomly selected n symbols from the dataset and computed the average precision and recall for these queries. This experiment has been repeated 100 times for the sake of stability and the averaged curves are presented in Figs 9a. First, we notice in the Figs. 9b that the changes in the number of classes affect different properties depending on the method. The recall of method a drastically decreases when introducing more and more symbol classes, whereas the precision of method c suffers much more than the recall. On the other hand, methods b and d seem to be equally affected by changes in scale in both precision and recall. From Figs. 9c we can see how the variations in the $F_A^1(r)$-score space are good indicators of the scalability of the methods under study. We can see in Table 3 the quality indicators for scalability tests. We present the mean of the standard deviations and its maximums. Again, methods b and d show much more scalability than methods c and a when looking at the composite measure. Finally, Figs. 10a and 10b show the scalability test at symbol level when

Table 3: Scalability test details.

Method	**Recall**		**Precision**		***F*-score**	
	$\overline{std_R}$	$\max(std_R)$	$\overline{std_P}$	$\max(std_P)$	$\overline{std_F}$	$\max(std_F)$
a	6.06	11.37	2.32	4.2	2.2	4.09
b	2.22	3.38	**1.74**	**2.01**	0.98	1.84
c	**1.02**	**1.6**	2.18	2.64	1.21	2.17
d	1.96	2.66	2.46	3.44	**0.85**	**1.09**



(a) Precision recall plots for several symbol classes.

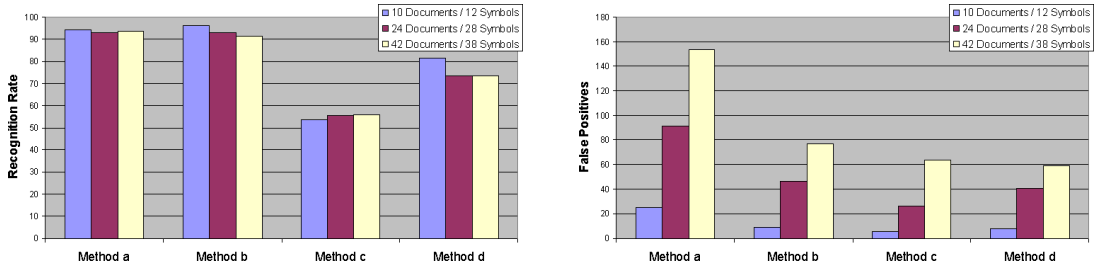
(b) Averaged precision and recall plots with standard deviations following the vertical (precision) and horizontal (recall) axes.

(c) Averaged $F_A^1(r)$-score plot with associated standard deviations.

Fig. 9: Scalability tests for all the four methods.

increasing both the number of models and the dataset size. As we can appreciate, the recognition rates vary slightly whereas the number of false alarms is exponentially increased in all the cases along with the dataset size.

From these results we can conclude that methods b and d seem to be much better than the other two. Method b should be chosen when we desire to retrieve as much symbols as possible, and on the other hand method d is suitable if we want to reduce the amount of false positives. Method a should only be chosen if the presence of false positives is not a problem and the user prioritizes finding all the positive symbols despite of the presence of false positives. However its performance seems to be affected by the number of considered symbols. Finally, method c is only suitable if we are interested in retrieving positive symbols at the first positions of the ranked retrieved locations even if we completely miss the rest of the symbols. Methods b and d also tolerate well changes in the number of considered classes and should be considered when facing applications involving a large amount of data. On the other



(a) Recognition Rates. (b) False Positives.

Fig. 10: Scalability tests at symbol level. Recognition rates (a) and false positives (b) are computed for three different scales of the dataset in terms of query models and documents in the collection.

hand, the strong points of methods *a* and *c* are compromised when introducing more and more symbol classes. All these conclusions are in accordance with the expected behavior of the studied methods, showing that the proposed evaluation protocol emphasizes the expected strengths and weaknesses of the methods under study.

6 Discussion and Conclusion

One of the main criticisms of using precision and recall to evaluate the performance of classification and location tasks is that it is sometimes difficult to really asses the behavior of the system under study. As claimed in [23], a low precision value can be due to a low accuracy in the recognition or to a bad localization due to over-segmenting. In addition, as pointed out in [41], the amount of overlap between polygons seems not to be a perceptively valid measure of quality. Quality indicators as the F-score have been also questioned, in [25] it is argued that this measure makes the systems look like they are much better than they really are.

We believe that the presented measures are able to evaluate well the behavior of symbol spotting systems, emphasizing their strong and weak points, and their tolerance to changes in scale. Precision is sometimes hard to interpret or does not provide perceptively good indicators, but the point of a spotting system is to retrieve zones of interest of document images, and the presented measures aim to measure the system's ability to do this task. Quality indicators aim to rank the methods according to certain ability, so even if the numbers by themselves do not have an accurate absolute meaning they are useful to compare methods between them. Finally, precision and recall are enhanced by measures working only at symbol level and the generality factor which helps to interpret the meaning of the plots. As shown in the evaluation section, the results obtained by using the proposed evaluation protocol are consistent with the ratios working at symbol recognition level, and most importantly, emphasizes the expected strengths and weaknesses of the methods under study.

Acknowledgements This work has been partially supported by the Spanish projects TIN2006-15694-C02-02 and CONSOLIDER-INGENIO 2010 (CSD2007-00018). The authors would also like to thank the anonymous reviewers for their helpful and constructive comments, as well as Dimosthenis Karatzas for proofreading the manuscript.

References

1. Aksoy, S., Ye, M., Schauf, M., Song, M., Wang, Y., Haralick, R.: Algorithm performance contest. In: Proceedings of the Fifteenth International Conference on Pattern Recognition, ICPR00, pp. 870–876 (2000). DOI 10.1109/ICPR.2000.903054
2. Antonacopoulos, A., Bridson, D.: Performance analysis framework for layout analysis methods. In: Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR07, pp. 1258–1262 (2007). DOI 10.1109/ICDAR.2007.4377117
3. Antonacopoulos, A., Gatos, B., Bridson, D.: ICDAR 2005 page segmentation competition. In: Proceedings of the Eighth International Conference on Document Analysis and Recognition, ICDAR05, pp. 75–79 (2005). DOI 10.1109/ICDAR.2005.184
4. Antonacopoulos, A., Gatos, B., Bridson, D.: ICDAR 2007 page segmentation competition. In: Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR07, pp. 1279–1283 (2007). DOI 10.1109/ICDAR.2007.203
5. Antonacopoulos, A., Gatos, B., Karatzas, D.: ICDAR 2003 page segmentation competition. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition, ICDAR03, pp. 688–692 (2003). DOI 10.1109/ICDAR.2003.1227750
6. Antonacopoulos, A., Karatzas, D., Bridson, D.: Ground truth for layout analysis performance evaluation. In: Document Analysis Systems, DAS06, *Lecture Notes on Computer Science*, vol. 3872, pp. 302–311. Springer Verlag (2006). DOI 10.1007/11669487_27
7. Barber, C., Dobkin, D., Huhdanpaa, H.: The quickhull algorithm for convex hulls. ACM Transactions on Mathematical Software **22**(4), 469–483 (1996). DOI 10.1145/235815.235821

8. Buckland, M., Gey, F.: The relationship between recall and precision. Journal of the American Society for Information Science **45**(1), 12–19 (1994)
9. Davis, J., Goadrich, M.: The relationship between precision-recall and ROC curves. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 233–240 (2006). DOI 10.1145/1143844.1143874
10. Delalandre, M., Pridmore, T., Valveny, E., Locteau, H., Trupin, E.: Building synthetic graphical documents for performance evaluation. In: Graphics Recognition. Recent Advances and New Opportunities, *Lecture Notes on Computer Science*, vol. 5046, pp. 288–298. Springer Verlag (2008). DOI 10.1007/978-3-540-88188-9_27
11. Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters **27**(8), 861–874 (2006). DOI 10.1016/j.patrec.2005.10.010
12. Holz, F., Witschel, H., Heinrich, G., Heyer, G., Teresniak, S.: An evaluation framework for semantic search in p2p networks. In: Proceedings of the Seventh International Workshop on Innovative Internet Community Systems, I2CS07 (2007)
13. Hripcsak, G., Rothschild, A.: Agreement, the F-measure, and reliability in information retrieval. Journal of the American Medical Informatics Association **12**(3), 296–298 (2005). DOI 10.1016/j.jamia.2005.01.008
14. Hu, M.: Visual pattern recognition by moment invariants. IRE Trans. on Information Theory **8**, 179–187 (1962)
15. Huijsmans, D., Sebe, N.: How to complete performance graphs in content-based image retrieval: add generality and normalize scope. IEEE Trans. on Pattern Analysis and Machine Intelligence **27**(2), 245–251 (2005). DOI 10.1109/TPAMI.2005.30
16. Kang, B., Kim, H., Lee, S.: Performance analysis of semantic indexing in text retrieval. In: Computational Linguistics and Intelligent Text Processing, *Lecture Notes on Computer Science*, vol. 2945, pp. 433–436. Springer Verlag (2004). DOI 10.1007/b95558
17. Kauppinen, H., Seppänen, T., Pietikäinen, M.: An experimental comparison of autoregressive and fourier-based descriptors in 2d shape classification. IEEE Trans. on Pattern Analysis and Machine Intelligence **17**, 201–207 (1995). DOI 10.1109/34.368168
18. Lambert, G., Gao, H.: Discrimination properties of invariants using the line moments of vectorized contours. In: Proceedings of the 13th International Conference on Pattern Recognition, ICPR96, pp. 735–739 (1996). DOI 10.1109/ICPR.1996.546920
19. Liu, W., Dori, D.: A protocol for performance evaluation of line detection algorithms. Machine Vision and Applications **9**(5), 240–250 (1997). DOI 10.1007/s001380050045
20. Liu, W., Dori, D.: Incremental arc segmentation algorithm and its evaluation. IEEE Trans. on Pattern Analysis and Machine Intelligence **20**(4), 424–431 (1998). DOI 10.1109/34.677280
21. Lopresti, D., Nagy, G.: Issues in ground-truthing graphic documents. In: Graphics Recognition Algorithms and Applications, *Lecture Notes on Computer Science*, vol. 2390, pp. 46–66. Springer Verlag (2001). DOI 10.1007/3-540-45868-9
22. Lu, C., Shukla, M., Subramanya, S., Wu, Y.: Performance evaluation of desktop search engines. In: Proceedings of the IEEE International Conference on Information Reuse and Integration, IRI07, pp. 110–115 (2007). DOI 10.1109/IRI.2007.4296606
23. Lucas, S.: ICDAR 2005 text locating competition results. In: Proceedings of the Eighth International Conference on Document Analysis and Recognition, ICDAR05, pp. 80–84 (2005). DOI 10.1109/ICDAR.2005.231
24. Lucas, S., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R., Ashida, K., Nagai, H., Okamoto, M., Yamamoto, H., Miyao, H., Zhu, J., Ou, W., Wolf, C., Jolion, J., Todoran, L., Worring, M., Lin, X.: ICDAR 2003 robust reading competitions: entries, results, and future directions. International Journal on Document Analysis and Recognition **7**(2), 105–122 (2005). DOI 10.1007/s10032-004-0134-3
25. Makhoul, J., Kubala, F., Schwartz, R., Weischedel, R.: Performance measures for information extraction. In: Proceedings of DARPA Broadcast News Workshop (1999)
26. Marcus, J.: A novel algorithm for HMM word spotting performance evaluation and error analysis. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP92, pp. 89–92 (1992). DOI 10.1109/ICASSP.1992.226113
27. Müller, H., Müller, W., Squire, D., Marchand-Maillet, S., Pun, T.: Performance evaluation in content-based image retrieval: overview and proposals. Pattern Recognition Letters **22**(5), 593–601 (2001). DOI 10.1016/S0167-8655(00)00118-5
28. Neumann, T., Bender, M., Michel, S., Weikum, G.: A reproducible benchmark for P2P retrieval. In: Proceedings of the First International Workshop on Performance and Evaluation of Data Management Systems, ExpDB06, pp. 1–8 (2006)
29. Phillips, I., Chhabra, A.: Empirical performance evaluation of graphics recognition systems. IEEE Trans. on Pattern Analysis and Machine Intelligence **21**(9), 849–870 (1999). DOI 10.1109/34.790427
30. Rath, T., Manmatha, R.: Features for word spotting in historical manuscripts. In: Proceedings of the Seventh International Conf. on Document Analysis and Recognition, ICDAR03, pp. 218–222 (2003). DOI 10.1109/ICDAR.2003.1227662
31. van Rijsbergen, C.: Information Retrieval. Butterworth-Heinemann Newton, MA, USA (1979)
32. Rusiñol, M., Lladós, J.: Word and symbol spotting using spatial organization of local descriptors. In: The Eighth IAPR International Workshop on Document Analysis Systems, DAS08, pp. 489–496 (2008). DOI 10.1109/DAS.2008.24
33. Smeaton, A., Over, P., Kraaij, W.: Evaluation campaigns and TRECVid. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330 (2006). DOI 10.1145/1178677.1178722
34. Song, J., Su, F., Tai, C., Cai, S.: An object-oriented progressive-simplification-based vectorization system for engineering drawings: Model, algorithm, and performance. IEEE Trans. on Pattern Analysis and Machine Intelligence **24**(8), 1048–1060 (2002). DOI 10.1109/TPAMI.2002.1023802
35. Stoyan, D., Stoyan, H.: Fractals, Random Shapes and Point Fields (Methods of Geometrical Statistics). John Wiley & Sons, Chichester (1994)
36. Tabbone, S., Zuwala, D.: An indexing method for graphical documents. In: Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR07, pp. 789–793 (2007). DOI 10.1109/ICDAR.2007.4377023
37. Tombre, K., Lamiroy, B.: Graphics recognition - from re-engineering to retrieval. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition, ICDAR03, pp. 148–155 (2003)
38. Valveny, E., Dosch, P.: Symbol recognition contest: A synthesis. In: Graphics Recognition, Recent Advances and Perspectives, *Lecture Notes on Computer Science*, vol. 3088, pp. 368–385. Springer Verlag (2004). DOI 10.1007/b99011
39. Valveny, E., Dosch, P.: Report on the second symbol recognition contest. In: Graphics Recognition. Ten Years Review and Future Perspectives, *Lecture Notes on Computer Science*, vol. 3926, pp. 381–397. Springer Verlag (2006). DOI 10.1007/11767978

40. Valveny, E., Dosch, P., Winstanley, A., Zhou, Y., Yang, S., Yan, L., Liu, W., Elliman, D., Delalandre, M., Trupin, E., Adam, S., Ogier, J.: A general framework for the evaluation of symbol recognition methods. International Journal on Document Analysis and Recognition **9**(1), 59–74 (2007). DOI 10.1007/s10032-006-0033-x
41. Wolf, C., Jolion, J.: Object count/area graphs for the evaluation of object detection and segmentation algorithms. International Journal on Document Analysis and Recognition **8**(4), 280–296 (2006). DOI 10.1007/s10032-006-0014-0