Combination of OCR engines for page segmentation based on performance evaluation*

Miquel Ferrer, Ernest Valveny
Computer Vision Center - Dept. Ciències Computació
Universitat Autònoma de Barcelona
Bellaterra, Spain
{mferrer,ernest@cvc.uab.es}

Abstract

In this paper we present a method to improve the performance of individual page segmentation engines based on the combination of the output of several engines. The rules of combination are designed after analyzing the results of each individual method. This analysis is performed using a performance evaluation framework that aims at characterizing each method according to its strengths and weaknesses rather than computing a single performance measure telling which is the "best" segmentation method.

1. Introduction

Page segmentation plays an important role in document analysis systems. It consists of decomposing a document image into its structural units such as regions or zones according to the type of data they contain. The zones are then labeled (or classified) with types such as text, graph, table, etc. Although many methods exist for page segmentation [2, 3, 6] neither of them can achieve perfect results in all situations. Following the same arguments used in other fields of pattern recognition for the combination of classifiers [4], one possibility to improve the performance of page segmentation is the combination of the results of different engines.

In this context, performance evaluation of page segmentation methods becomes a very important and useful issue. If we want to define strategies for the combination of segmentation engines, we need to characterize them in order to know their strengths and weaknesses, according to their output. Recent research works [5, 7, 8] and segmentation competitions [1] have proposed different frameworks to evaluate the performance of page segmentation output. Most of them define a global score indicating the accuracy of page segmentation engines, based on the combination of different types of errors [8] or measures[7]. In [1] two basic measures called "Entity Detection Metric" and "Segmentation Metric" are given based on existing measures in [7].

These methods are not able to characterize the behavior of a page segmentation engine in terms of the type of errors they make and the accuracy they achieve in both segmentation and classification of zones. Thus, instead of a single metric, we propose to define a set of metrics that characterize the behavior of a page segmentation engine when segmenting complex documents such as magazines or newspapers where there are multiple types of zones and the zones themselves are difficult (non-Manhattan style). Our goal is to define a set of measures which can help in determining the strengths and weaknesses of each segmentation engine, i.e, identifying the best engine for the detection of a particular type of zone. Based on the results of this evaluation framework we can analyze the behavior of each method and propose specific combination rules that improve the segmentation obtained by each individual engine. In this paper, we will show such strategy for a particular case using two commercial engines and a set of pages from a magazine.

In this paper we start by describing in section 2 the framework that permits an automatic evaluation and comparison of different segmentation engines. First, we make some considerations about the generation of the ground-truth that leads us to introduce a new type of zone, what we have called *text over image (ToI)*. Then, we explain the set of evaluation measures defined in order to compare the ground-truth and the results of the OCR engines. In section 3 we use this framework to evaluate a particular scenario where we apply two commercial engines for the segmentation of a set of pages from a magazine. The results of the evaluation are analyzed and used to design two specific combination rules that are shown to improve the results of the segmentation. Finally, in section 4 we state the main conclusions of this work.

*This work was sponsored by Fellowship number 401-027 (UAB) / Cicyt TIN2006-15694-C02-02 (Ministerio Ciencia y Tecnologa).

2. Evaluation framework

Two main issues should be considered when designing any evaluation framework: the definition of the ground-truth and the definition of the performance measures used to evaluate the methods. In the following, we will explain how we have defined both issues taking into account our motivation: the improvement of the overall performance of a set of segmentation engines through their combination.

2.1. Ground-truth

For a real performance evaluation of page segmentation algorithms, it seems clear that the ground-truth must be based on scanned real images of documents. That is the approach we have taken and we have selected a set of pages from magazines, containing various types of zones and layouts. Then, the main problem is the annotation of such images. This annotation has to be done manually by users and, therefore, a lot of ambiguity can be generated concerning the limits and types of the zones.

In order to reduce this ambiguity we have defined a set of standard rules that have to be followed by any user, but we have also introduced a new type of zone that it is not usually considered. This new type of zone is defined in order to handle situations such as that illustrated in figure 1 where we have an image containing text. In these cases, some engines label the zone as *text*, while others can label it as *graphics* and even others can give two zones labelled as *text* and *graphics* respectively. Then, whatever the label we assign in the ground-truth, we could generate errors in the results of segmentation. Thus, we have introduced a new type of zone, called *text over image* to label this kind of situations. Thus, whatever the output of the segmentation engine is (text or graphics) we can consider it as correct.

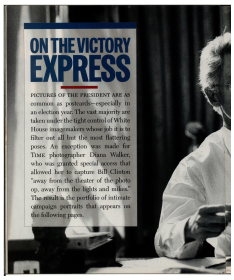


Figure 1. Example of a *Text Over Image* zone.

Since segmentation engines do not use this new type of zone, the *text over image* category is artificially generated when we analyze the output of the engines. The goal is to detect zones or parts of a zone which are labelled as *text* and overlap with *graphics* zones. Then, the intersection between them is labelled as *text over image* and is subtracted from original text and graphics zones (see figure 2).

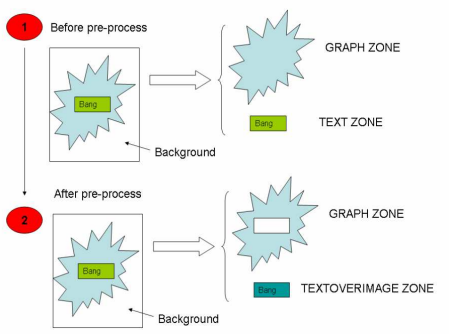


Figure 2. Creation of *Text Over Image* zones.

Summarizing, we use five types of categories to label the zones in the ground-truth: the three usual categoires given by OCR engines: *text*, *graphics* (for graphics and images) and *table*, the new category *text over image* and *background* (anything which does not fall into previous categories).

2.2. Performance evaluation

We recall that the main goal of our evaluation framework is to characterize the performance of each segmentation in order to combine them according the result of this evaluation. Therefore, we are not interested in defining a single measure of global performance, telling which is the "best" segmentation algorithm, but in defining a set of measures that help to understand in which situations a given engine works better than the others. This information must guide the definition of combination rules.

This set of measures will be computed with the help of a table that allows to match the ground-truth zones with the zones given by the segmentation algorithm, in a similar way as in [7]. This table has a number of rows equal to the number of zones in the output (S-zones) and a number of columns equal to the number of zones in the ground-truth (GT-zones). One row and one column are added to this table in order to deal with the background zones in the output and the ground-truth respectively. Each cell in the table contains information about the intersection between a zone in the output and a zone in the ground-truth. Once the table has been filled, we compute for every ground-truth region how it is distributed or shared among output zones (we call this GT-results). Then, the same procedure is repeated for every output zone (we call this S-results) –see figure 3.

The GT- Results and S-Results columns contain information about what percentage of every zone in the ground-truth/output has been recognized/assigned, how many zones fall into this zone and their types, etc. A zone can be totally, partially or not recognized/assigned. If it was totally or partially recognized then it could be identified only with zones

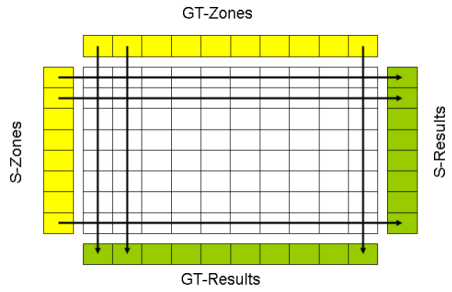


Figure 3. Matching table between ground-truth and output zones.

with the same type, only with zones with different type or with zones with both the same type and different type. This information will be used to compute the set of performance measures. These measures are grouped into 6 categories:

Correct recognition: include percentages of ground truth area (grouped by zone type), that have been correctly recognized as zones with the same type.

Unrecognition: report percentage of zones labelled as text, graph, text over image or table in the ground truth and recognized as background by a an engine.

Missrecognition: these measures are intended to detect zones in the ground truth that have been recognized as a different type in the output. For example a text zone recognized as graphics, text over image or table.

Overlap: compute the area (grouped by type) in the ground-truth that is recognized twice by the zoning engine. The result is relative to the total area of every type recognized by the zoning engine. This sometimes happens especially in engines that produce Manhattan style zones only.

Split and merge: tell us how many zones in the ground-truth are splited/merged in several zones or a single zone respectively in the output.

3. Combination of segmentation results

Once the results for each engine are obtained and their behavior is characterized using the set of measures we presented before, we are able to infer a set of combination rules of their outputs in order to improve the individual results achieved by each engine. In this section we will first present the evaluation experiment we have done over each individual engine. Then we will present two sets of combination rules based on the results obtained, and finally we will present the results obtained using such combination rules comparing them with those obtained by the engines.

3.1. Evaluation experiment

The dataset used in this experiment was composed of 92 images of pages scanned from a well-known magazine. The groundtruth for each page was manually defined as explained before. For each page, the zones and their type were stored in a XML file. Then, every original image was used as an input for each engine, obtaining, for every page, the results of their segmentation as another XML file containing the detected zones and their labels. After that, the results of each engine were compared against the ground-truth in the way explained in section2, obtaining the complete set of performance measures.

In the two first columns of table 1 we can see a summary of the measures of *correct recognition*, *missrecognition* and *unrecognition* for each engine. This summary shows the total percentage of area of text, graphics and text over image, respectively, that has been correctly recognized, missrecognized as other types of zones or unrecognized for every engine. The best result of correct recognition for each type of zone has been highlighted. The main conclusion that we can draw from the analysis of this table is that *engine 1* achieves a very good performance in text detection while it is not able to recognize a significant percentage of graphics (almost 30%). On the contrary, *engine 2* has a good percentage of graphics detection, but a high rate of text missrecognition (almost 40%). In a more deep analysis we have seen that most of these missrecognized graphics zones are missrecognized as text. These conclusions will be used in the next section to design the combination rules.

In addition to these quantitative results, we generated a more qualitative results which consists of adding a semi-transparent color zones to the original images showing the differences between the ground-truth and the output of the engine. An example of such images is shown in figure4. In this case zones surrounded by a wide white line correspond to missrecognized, unrecognized or overlapped zones, either text or graphics.

3.2. Definition of combination rules

In this section, we present two sets of combination rules that were inferred based on the results of previous section.

3.2.1 Combination 1

This combination is composed of a simple and straightforward set of rules. Taking into account the Text and Table good recognition measures of the *engine 1* and the Graph good recognition of *engine 2*, the set of rules for this combination was:

1. All the zones labelled as *text* by engine 1 were included in the combination.

2. All the zones labelled as *table* by engine 1 were included in the combination.

3. All the zones labelled as *graphics* by engine 2 were included in the combination.

So, summarizing we can say that this combination was composed by the *text* and *table* zones of the first engine and the *graphics* zones of the second engine, discarding the *graphics* zones of the first engine and the *text* and *table* zones of the second engine.

3.2.2 Combination 2

After that, a set of more complex rules were defined in order to try to slightly improve the results obtained with the first combination. In the following the set of rules for this second combination are presented:

1. All the zones labelled as *text* by engine 1 were included in the combination

2. All the zones labelled as *table* by engine 1 were included in the combination

3. The zones recognized as graph in the engine 1 that have graph overlap less than a certain threshold with the zones of engine 2 were included to the combination as graph zones.

4. The zones recognized as graph by engine 2 that overlap only with one text zone and the overlap is greater than 80% were not included in the combination.

5. The rest of *graphics* zones produced by engine 2 were included in the combination

6. All the zones labeled as *ToI* in engine 2 were included as *text* zones in the combination

This set of rules were defined after visual inspection of the results obtained with the first combination. All thresholds were selected in order to maximize the global rates of zone detection. Notice that rules 1, 2 and 5 are almost identical to the rules of the first combination. Rule 3 comes from the evidence that some of the unrecognized *graphics* zones in the first combination were detected by the first engine but not for the second. For this reason this new rule including the graph zones that overlap with graph zones detected by the second engine up to a certain threshold (in this case 30%) were added. In addition, the resulting images of the first combination also revealed us that some missrecognized *graphics* zones detected by the second engine were good recognized correctly as text zones by the first engine. These *graphics* zones were included in the first combination leading to a *ToI* zone in the combination where *text* zone should be detected. For this reason, these *graphics* zones of the second engine were excluded in the combintation (rule 4). Finally, some *ToI* zones detected by the second engine were not found by the first engine. The second rule corrects this error by adding the *ToI* zones found by the second engine as *text* zones in the final combination (rule 6).

3.3. Results of the combination

The results obtained for the combinations explained before are summarized in tables 1 and 2. Concerning *combination 1* we can see that the rate of correct graphics recognition is almost the same than the rate obtained by *engine 2*, the best engine in graphics detection. At the same time, this combination permits to increase the recognition of text over image as we overlap correct text zones detected by *engine 1* over correct graphics zones detected by *engine 2*. Even if the rate of correct text recognition is lower than the rate for *engine 1*, we can see in table 2 that the missing text has been recognized as text over image, which means that the combination keeps correct text zones of *engine 1* but overlaps them with missrecognized graphics zones of *engine 2*.

Table 1. Results obtained for the engines (E1 and E2) and the combinations (C1 and C2).

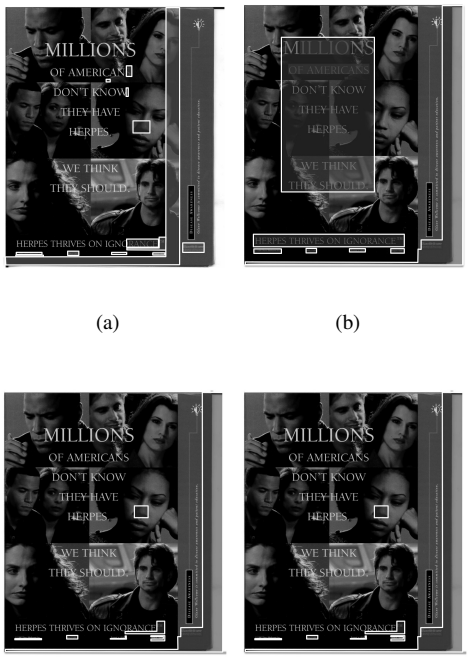
	E1	E2	C1	C2
Text good recognition	**94.51**	55.85	79.15	86.28
Text missrecognition	4.07	37.99	20.17	13.08
Text unrecognition	1.45	7.30	0.77	0.74
Graphics good recognition	67.20	84.65	83.32	**90.67**
Graphics missrecognition	3.05	2.65	3.05	3.09
Graphics unrecognition	29.75	13.21	13.74	6.45
ToI good recognition	13.17	6.50	43.90	**47.84**
ToI missrecognition	51.18	86.33	50.15	46.73
ToI unrecognition	35.65	7.31	5.95	5.64

Table 2. Text Missrecognition results

	E1	E2	C1	C2
Text Recognized as Graphics	2.52	16.27	3.28	3.30
Text Recognized as ToI	1.16	3.42	16.49	9.39
Text Recognized as Table	0.39	18.30	0.39	0.39

In the case of *combination 2* we can see from table 1 that we increase significantly the rate of graphic zones good recognition which means that with the new rules we are able

to add graphic zones detected by *engine 1* and not detected by *engine 2*. In addition, we also achieve an increase in the rate of correct text recognition because we reduce the overlap between text zones of *engine 1* and graphic zones of *engine 2*. Finally we are also able to improve a little the rate of text over image recognition. As a final conclusion, we want to remark that adding the text recognized as text over image in *combination 2* we obtain rates of text, graphics and text over image recognition better than the best individual engine for each type of zone.



(a) (b)

(c) (d)

Figure 4. Error produced by engines 1 (a) and 2 (b), and combinations 1 (c) and 2 (d).

In order to provide a qualitative measure of these results, we include a visual example of the results obtained in one page by the two engines and also by both combinations. In image 4 zones with white borders correspond to groundtruth zones not recognized or missrecognized by each of the engines or combination schemes. We can see how, *combination 2* obtains more accurate results than *combination 1* and that the final result of the combination is comparable to the result of the best individual engine for each type of zone.

4. Conclusions

In this work we propose the combination of the output of several page segmentation engines in order to improve the performance achieved by each method individually. The combination is defined based on the performance evaluation of the segmentation engines. To this end, we have developed a performance evaluation framework whose main goal is to characterize each engine according to the types of zones that it is able to detect, instead of giving a single measure trying to determine the best method. In this sense, we have introduced a new type of zone in the ground-truth, what we have called text over image, in order to reduce the ambiguity when labelling text zones that overlap graphic zones. As a result, we compute a set of measures that permit to obtain the rates of correct recognition, missrecognition and unrecognition for each type of zone.

For the experimental evaluation of this framework we have used two commercial OCR engines in order to obtain the segmentation of 92 pages of a well-known journal. These results have been analyzed with the evaluation framework and used to design two different sets of combination rules. The results show that the use of these combination schemes improve the results of each individual engine. In particular, the second combination permits to obtain a final segmentation output that is better than the output of best individual engine for each type of zone.

Although these combination rules have been designed ad-hoc for a particular case of application, this approach could be generalized to other situations. Future work will be devoted to the automatic inference of a set of combination rules given the results of the evaluation of different engines on a test set.

References

[1] A. Antonacopoulos, B. Gatos, and D. Bridson. Icdar2005 page segmentation competition. In *Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR 2005)*, pages 75–79, August 2005. Seoul, Korea.

[2] A. K. Jain and B. Yu. Document representation and its application to page decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(3):294–308, 1998.

[3] D. P. d. James L. Fisher, Stuart C. Hinds. A rule-based system for document image segmentation. *ICPR*, pages 567–572, 1990.

[4] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Trans. PAMI*, 20(3):226–239, 1998.

[5] S. Mao and T. Kanungo. Software architecture of pset: A page segmentation evaluation toolkit. *International Journal on Document Analysis and Recognition*, 4(3):205–217, 2002.

[6] L. O'Gorman. The document spectrum for page layout analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(11):1162–1173, 1993.

[7] I. T. Phillips and A. K. Chhabra. Empirical performance evaluation of graphics recognition systems. *IEEE Trans. PAMI*, 21(9):849–870, 1999.

[8] B. Yanikoglu and L. Vincent. Pink panther: a complete environment for ground-truthing and benchmarking document page segmentation. *Pattern Recognition*, 31:1191–1204, September 1998.