From text detection to text segmentation: a unified evaluation scheme

Stefania Calarasanu[1], Jonathan Fabrizio[1] and Séverine Dubuisson[2]

[1]EPITA-LRDE, 14-16, rue Voltaire, F-94276, Le Kremlin Bicêtre, France,
{calarasanu,jonathan.fabrizio}@lrde.epita.fr
[2]Sorbonne Universités, UPMC Univ Paris 06, CNRS, UMR 7222, ISIR, F-75005, Paris, France
severine.dubuisson@isir.upmc.fr

Abstract. Current text segmentation evaluation protocols are often incapable of properly handling different scenarios (broken/merged/partial characters). This leads to scores that incorrectly reflect the segmentation accuracy. In this article we propose a new evaluation scheme that overcomes most of the existent drawbacks by extending the EvaLTex protocol (initially designed to evaluate text detection at region level). This new unified platform has numerous advantages: it is able to evaluate a text understanding system at every detection stage and granularity level (paragraph/line/word and now character) by using the same metrics and matching rules; it is robust to all segmentation scenarios; it provides a qualitative and quantitative evaluation and a visual score representation that captures the whole behavior of a segmentation algorithm. Experimental results on nine segmentation algorithms using different evaluation frameworks are also provided to emphasize the interest of our method.

Keywords: Evaluation protocol, evaluation metrics, text segmentation.

1 Introduction

During the last decade, text understanding systems have received a lot of attentions from both the research and the industry communities. In particular, end-to-end systems became popular due to their complex processing chain. Such systems rely on different processing stages, such as text segmentation, text grouping, text classification, text localization, text rectification or text recognition. Among these stages, text segmentation is a phase of crucial importance not only for many end-to-end systems but also for other applications that rely on it, such as the image inpaiting (*e.g.* subtitle removal [21]). The interest in this stage is also reflected by the organization of numerous competitions around this topic, such as ICDAR 2013 [15] and 2015 [13] *Robust Reading Competition* (Task 2) with Challenge 1 on born-digital images and Challenge 2 on natural scenes.

The different stages of an end-to-end system are evaluated in different ways, with respect to the type of text representation at each level. For example, text segmentation implies evaluating a binary representation, while text localization is usually done by comparing bounding box positions in an image. Finally, the text recognition is evaluated based on the transcription obtained by using an OCR. Frequently, the evaluation of the transcription is also used as a final result for many end-to-end systems. This result provides a combined evaluation based on the recognition accuracy of the used OCR and the localization precision. Since text segmentation can be a vital pre-processing or post-processing stage in a document analysis application, it is important to quantify its contribution and quality independently.

Most of the times the output of the text segmentation is a binary image in which the foreground (white pixels) represents potential text objects (*i.e.* characters) and the background (black pixels) corresponds to the remainder.

As pointed in [20], among the existing evaluation choices (user-interaction, output OCR, pixel-based...) the most valuable evaluation procedure seems to be the pixel comparison between the ground truth (GT) and detection images. Nevertheless, evaluating text segmentation remains a difficult task due to a set of problems that can alter the evaluation scores such as: variation of characters' thickness, merged characters, partially detected characters or fragmented characters. For example, the variation of characters' thickness can often impact the evaluation scores when dealing with natural scene images, in which illumination properties and cluttered backgrounds can severely influence the accuracy of the character segmentation. We note that, due to these difficulties, some protocols have common drawbacks, the three most penalizing ones being:

- In pixel-based evaluation protocols, the penalty due to a character's segmentation failure varies usually with respect to the character's surface size, although the surface is not related to the size of the character.
- Conversely, many character-based evaluation protocols imply a binary approach to decide whether a character is well segmented or not; this leads to inaccurate evaluations of degraded character segmentations.
- Lastly, dealing with binary decisions implies a threshold dependency of critical parameters which are most of the times difficult to set up.

In this article we propose to overcome these issues by firstly adapting the EvaLTex protocol exposed in [2] initially designed for text localization and secondly using the visualization tool in [1], to evaluate text segmentation results. The advantages of the adaptation of this framework are numerous:

- It provides a more representative set of scores than classical evaluation methods.
- It comes with a visualization system which gives at a glance an overview of the accuracy of the evaluated methods and helps to compare different sets of results.
- It handles all types of atom detections: partial, broken, merged, broken and merged, missed.
- It produces an equal evaluation of all characters regardless of their size.
- It proposes a non-binary local evaluation which allows a proper evaluation of partial, broken and merged character detections.

– It is robust to character thickness variation.

Furthermore, the EvaLTex protocol, by design, is able to evaluate text detection systems at paragraph/line/word level. The adaptation of EvaLTex protocol for text segmentation unifies the evaluation at all detection levels: paragraph/line/word and now character. It is hence possible to evaluate an end-to-end system, at each step, by using a single tool.

This article is organized as follows. Section 2 presents a survey of the state-of-the-art evaluation methods used for text segmentation. This section also presents the EvaLTex protocol that is next adapted for the segmentation evaluation. Section 3 presents the evaluation framework details and its applicability to text segmentation evaluation tasks. Section 4 is dedicated to experiments and discussions on the scores obtained with different evaluation methods. Finally, concluding remarks and perspectives are given in Section 5.

2 Related work

Text segmentation can be evaluated in different manners. For example, by user-interaction [25,26,4,12,6], in which humans are supposed to count manually the number of correct and incorrect matches between the GT and the detections. This approach is not sufficiently reliable as it inevitably implies a high level of subjectivity: different users can produce different evaluation scores.

Another approach for text segmentation evaluation consists in using the OCR recognition rate [10,27,3]. In such cases, the scores do not only depend on the segmentation correctness but also on the OCR recognition accuracy. For example, a good segmentation output can produce a low score if the used OCR does not recognize correctly all the characters.

The pixel-based evaluation methods [22,20,8,23,16,30] compute the difference between a binary GT image and a detection image and count the number of pixels that correctly match. In [22] five performance measurements were used to evaluate the binarization outputs: misclassification error (ME), edge mismatch (EMM), relative foreground area error (RAE), modified Hausdorff distance (MHD), and region non-uniformity (NU). Authors in [8] opted for recall, precision, accuracy and specificity rates to interpret historical documents binarization results. A normalized cross-correlation value between binarized and GT images has been employed in [16] to evaluate the degraded character level in color images. The main drawback of these approaches is the fact that the evaluation is done exclusively at pixel level and hence it depends on the character area size making it less robust to thickness variations. In [23], the authors presented an evaluation protocol for binarization methods, based on four metrics: pixel error ($PERR$), square error (MSE), signal to noise ratio (SNR) and peak signal to noise ratio ($PSNR$). The advantage of this method is that it comes together with a GT annotation technique consisting of adding noise to a clean document and hence assuring an objective pixel-based evaluation. In [30], authors also opted for the PSNR metric, together with a negative rate metric

(NRM) and a misclassification penalty metric (MPM). In [20] the evaluation is also done at pixel level by producing a set of eight evaluation metrics computed directly on binarized images: F-measure, recall, precision, broken text, missing text, merge-deformation, deformation and false alarm. The advantage of this work is that it provides a method to produce a reliable GT by using skeletons. Hence, the method is not sensitive to character thickness variations and does not alter the scores. This protocol was used during the ICDAR 2015 *Robust Reading* competition (Challenge 1 and 2, Task 2).

The advantage of this work is that it provides a method to produce a reliable GT by using skeletons. Hence, the method is not sensitive to character thickness variations and does not alter the scores. This protocol was used during the ICDAR 2015 *Robust Reading* competition (Challenge 1 and 2, Task 2).

Finally, the atom-based evaluation [5,9,2] consists in comparing atom-level objects (*i.e.* characters). Compared to pixel-based evaluations, this kind of approaches treats all characters in the same way, independently of their sizes. This is a major advantage with respect to pixel-based evaluations, because it can differentiate between various segmentation scenarios. Clavelli *et al.* [5] proposed a multi-level annotation scheme that represents text objects at pixel (text part), atom (e.g character), word and line levels. This framework, also used for the ICDAR 2015 *Robust Reading* competition (Challenge 1 and 2, Task 2) can evaluate text segmentation tasks, when text objects are represented at pixel and part levels. The matching protocol is based on two thresholds: $T_{\min}$ and $T_{\max}$, used to validate the matchings between a GT and a detection represented by a set of connected components (CCs). The default values are set to: $T_{\min} = 0.9$, $T_{\max} = \min(5, 0.5 \cdot T)$, where T corresponds to the thickness of the text part. Based on this, the detection CCs are classified into several categories: background, fraction, whole, multiple, fraction & multiple and mixed. The main disadvantages of this method are the binary local evaluation approach and the fact that it does not handle broken character segmentations, which leads to low recall values when such cases occur. The ZoneMap metric proposed in [9] is a generalization of the metric proposed in [18] and of the DETEVAL framework [28] used for evaluating page segmentation and area classification in documents. It is computed based on the matching scenarios and different error rates. EvaLTex [2] is a framework introduced to evaluate text localization results. The main core of this protocol consists of a two-level ground truth annotation for each image: first, each word is bounded by a rectangular box; then, several words are grouped and bounded into text regions. This two-level annotation is then used to compare the GT text objects with the detection results. Based on the overlap between the GT and the detection objects, four types of matchings are identified: *one-to-one*, *one-to-many*, *many-to-one* or *many-to-many*. Depending on the matching type, a dedicated set of local performance metrics for each GT object is computed. Finally, seven global scores (global, quantity and quality recall and precision and F-Score) are computed for a dataset, by providing both a quality and a quantity evaluation of the detection results. This protocol represents the starting point for the evaluation scheme presented in this paper.

3 Revision of EvaLTex and adaptation to text segmentation

In this section we show how, by respecting and adapting some of the metrics and rules of EvaLTex, we can evaluate not only text localization, but also text segmentation.

GT annotation. In order to evaluate a result, a GT is needed. For the character segmentation evaluation, we will consider the annotation representation to be a mask for a connected component (CC) or a set of CCs having the same label in the segmented image. In practice, in the Latin alphabet, a character is usually defined by one CC but it can also be defined by two or three CCs (*i.e.* characters with accents or tittles). An example of the used ground truth annotation is given in Fig. 1. In the following section we will discuss the matching rules and the metrics used to evaluate the text segmentation.

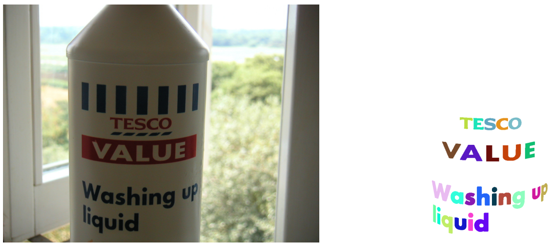


Fig. 1: Example of a text segmentation image with labeled CCs.

Matching strategies. Initially, the EvaLTex framework was designed to evaluate the localization of text regions such as words and lines represented through bounding boxes or blobs. It relies on the computation of two measurements: the coverage and the accuracy. The coverage measures the ratio between the matching surface of two objects with respect to the GT object, while the accuracy measures the ratio of the matching surface and the detection one. In order to provide a representative evaluation, these two measurements take into account many different detection situations. Simple cases involve an object from the GT being matched with a single object in the detection (*one-to-one* match). More challenging cases include GT objects being matched with multiple detections (*one-to-many*) or multiple detections matching the same object in the GT (*many-to-one*). Finally, it is possible that many objects in the GT match many objects in the detection (*many-to-many*). As the EvaLTex framework is able to handle all these cases and as it does not perform a binary evaluation but always provides a ratio of quality of the matching, the evaluation is more representative than many other frameworks that do not handle all different cases or simply perform a binary evaluation (match or failure). Furthermore, EvaLTex framework always considers slight variations of detected text elements by enlarging and reducing their surrounding regions during the comparison.

In order to apply the principles of EvaLTex to evaluate text segmentation one needs to move from the word/line/region representation to a character level using a connected component (CC) annotation. Similarly to text detection, text segmentation requires managing different matching scenarios. When a GT CC corresponds to a CC in the detection set, even if the coverage between the two is not complete, we deal with a whole or partial atom detection. When a CC in the GT is covered by multiple CCs in the detection set, we deal with a fragmented atom detection. The third case, which consists in multiple characters in the detection being linked together, is referred to a merged atom detection where a CC in the detection set corresponds to multiple CCs in the GT. Lastly, a fragmented and merged detection occurs when detections are fragmented and linked to other characters at the same time. One can observe that the matching cases that occur in text detection can be retrieved in the segmentation scenarios, making EvaLTex a good starting point to evaluate text segmentation algorithms. To do so, let us consider $\mathcal{G} = \{G_i\}_{i=1,\ldots,N_G}$ a set of N_G GT text boxes and $\mathcal{D} = \{D_j\}_{j=1,\ldots,N_D}$ a set of N_D detections.

Whole and partial atom detection. To locally evaluate the quality of the matching between a GT atom G_i and a detection D_j we can use the *coverage* and the *accuracy* metrics defined in [2] (this corresponds to a *one-to-one* case). The coverage value is computed using the reduced GT object, while the accuracy using the enlarged GT object in order to remain robust to small detection size variations. When moving from the evaluation of words to characters, we want to keep this property by remaining robust to slight character thickness variations. However, applying the erosion could make small characters disappear. Hence, we will consider the computation of the coverage value, not by eroding the GT object but by dilating the detection atom. This change will further be reflected in the evaluation of all four types of atom detections. For a *partial* or *whole* segmented CC, the coverage and accuracy between G_i and D_j is computed in the following manner:

$$\mathrm{Cov}_i = \frac{\mathrm{Area}(G_i \bigcap Dd_j)}{\mathrm{Area}(G_i)}, \quad \mathrm{Acc}_i = \frac{\mathrm{Area}(Gd_i \bigcap D_j)}{\mathrm{Area}(D_j)}, \tag{1}$$

where Gd_i and Dd_j represent the dilated GT and detection CCs. In our experiments we use a square structuring element of size equal to 1.

Fragmented atom detection. In the case of a *fragmented* CC (corresponding to a *one-to-many* detection scenario), the same GT CC is detected multiple times. Here again, the coverage [2] is computed by taking into account all the intersections between the GT CC and the dilated detection CCs in the following manner:

$$\mathrm{Cov}_i = \frac{\bigcup_{j=j_1}^{j_{s_i}} \mathrm{Area}(G_i \bigcap Dd_j) - \bigcap_{j=j_1}^{j_{s_i}} Dd_j}{\mathrm{Area}(G_i)} \cdot F_i, \tag{2}$$

$$\mathrm{Acc}_i = \frac{\bigcup_{j=j_1}^{j_{s_i}} \mathrm{Area}(Gd_i \bigcap D_j) - \bigcap_{j=j_1}^{j_{s_i}} Dd_j}{\bigcup_{j=1}^{s_i} \mathrm{Area}(D_j)} \tag{3}$$

where s_i represents the number of fragmentations associated to G_i and F_i a fragmentation penalization applied to each GT CC. Contrary to the penalization proposed in [2], we propose here a smoother one, defined as:

$$F_i = \frac{1}{1 + \ln(s_i) \cdot \ln(s_i)} \cdot 0.6 + 0.4 \tag{4}$$

Merged atom detection. A merged atom case (equivalent to a *many-to-one* scenario) considers the coverage rate used during partial and whole atom detection (see Eq. 1), while the accuracy is computed as in [2]:

$$\text{Acc}_i = \frac{\text{Area}(Gd_i \bigcap D_j)}{\text{Area}(D_{j,i})}, \tag{5}$$

where Area($D_{j,i}$) represents the corresponding detection area for each G_i and is defined as:

$$\text{Area}(D_{j,i}) = \frac{\text{Area}(Gd_i)}{\text{TextArea}_{D_j}} \cdot \text{nonTextArea}_{D_j}, \tag{6}$$

where TextArea_{D_j} and nonTextArea_{D_j} correspond to, respectively the total text area and the non text area, defined as:

$$\text{TextArea}_{D_j} = \text{Area}(\bigcup_{i=1}^{m_j}(Gd_i \bigcap D_j)) \tag{7}$$

$$\text{nonTextArea}_{D_j} = \text{Area}(D_j) - \text{TextArea}_{D_j}, \tag{8}$$

where m_j represents the number of merged GT CCs.

Fragmented and merged. The coverage for the fragmented and merged scenarios (corresponding to *many-to-many* cases) is derived from the fragmented case (see Eq. 2). In [2], the accuracy is computed exclusively using Eq. 5. However, since this case involves a merged and fragmented atom detection, we redefine the accuracy and compute it by combining Eq. 3 and 5. The accuracy then becomes the ratio between the union of all intersection areas between the GT CC G_i and the union of all detections k_i CC that are generated from the *merged* mappings as well as all s_i CC detections generated from the *whole* or *partial* mappings:

$$\text{Acc}_i = \frac{\bigcup_{j=j_1}^{j_{s_i+k_i}} \text{Area}(Gd_i \bigcap D_j) - \bigcap_{j=j_1}^{j_{s_i}} D_j}{(\bigcup_{j=j_1}^{j_{s_i}} \text{Area}(D_j) - \bigcap_{j=j_1}^{j_{s_i}} D_j) \, \bigcup \, (\bigcup_{j=j_1}^{j_{k_i}} \text{Area}(D_{k,i}))} \tag{9}$$

4 Experiments and interpretation of results

In this section we evaluate our evaluation scheme and compare our results with other evaluation protocols. This evaluation is performed on the segmentation results[1] of nine detection methods that participated to the *ICDAR 2013 and 2015 Robust Reading Competition* [15,13]. The EvaLTex tool and the corresponding GT format for the ICDAR datasets are available at `https://www.lrde.epita.fr/wiki/Evaltex`.

[1] Publicly available at `http://rrc.cvc.uab.es/` [13]

Evaluating an evaluation method is neither an easy, nor an obvious task. First of all, there are no precise rules that can decide the precision or correctness of such a protocol. Moreover, there is undoubtedly, always a level of subjectivity involved in the proposition of such a new protocol, either due to its metrics or its matching strategies. This raises a straightforward question of whether we can state that one evaluation protocol is better than other. While this statement seems too strong, we can however debate on the reliability of some metrics and computed scores in a given context. To validate at best our protocol, we first propose a qualitative evaluation (Sec. 4.1) using three evaluation methods: ours, one pixel-based [20] and one atom-based [5]. Next (Sec. 4.2), global scores on the entire dataset are computed using the different evaluation methods and compared. Lastly, in Sec. 4.3 we show the suitability of the histogram visualization tool to provide, at a glance, a more detailed overview of the behavior of segmentation methods.

4.1 Qualitative evaluation

The purpose of having a qualitative comparison is to stimulate the analysis of concrete examples and interpret the representativity of certain scores with respect to others in order to illustrate possible inconsistencies and their impact on the scores produced by current used protocols. Fig. 2 illustrates multiple common segmentation situations that will be used as a basis for further discussions. In this figure, from top to bottom, the first picture is the original image, the second picture is the expected segmentation (the GT) while the third represents the segmentation result. The next images correspond to the analysis of a pixel based evaluation, followed by an atom based evaluation analysis[2]. The sixth image corresponds to the histogram visualization of coverage and accuracy distributions for each segmentation example. Table 1 provides the scores computed using the atom-based, pixel-based and our evaluation method.

In Fig. 2.a one can observe that all text characters have been segmented. However, due to hole filled characters (“o”, “d”, “D” and “O”) the atom-based evaluation does not consider the characters as segmented (in red) and underestimates the recall (78%). As the evaluation is a binary one, the computed recall for this image is the same as if these letters would have been completely missed. The first conclusion is that such an evaluation does not allow precise comparisons. Moreover, the detected letters are considered as false positive which produces similar precision scores (70%) which is clearly underestimated. The atom-based protocol considers better to miss the “o” than to detect it with its hole filled. The second conclusion is that the obtained scores with this framework are not a faithful representation of what we would intuitively expect.

Fig. 2.b illustrates an example of merged characters (“U” and “T”), which are correctly identified by the atom-based evaluator, but not taken into account when computing the recall value. This leads to a recall of 60%, which is clearly not consistent with the segmentation efficiency. In the example depicted

[2] All pictures publicly available at `http://rrc.cvc.uab.es/` [13]

Table 1: Recall, Precision and F-Score obtained using different evaluation methods on the segmentation examples depicted in Fig. 2.

	Pixel [20]			Atom [5]			Our evaluation		
Fig.	R	P	F	R	P	F	R	P	F
2.a	94.04	46.22	61.98	77.78	70	73.68	99.96	83.72	91.12
2.b	92.18	57.25	70.63	60	42.86	50	99.08	53.19	69.22
2.c	82.96	93.75	88.02	66.67	69.23	67.92	72.12	79.79	75.76
2.d	92.9	68.05	78.56	66.67	75	70.59	88.88	79.24	83.79

in Fig. 2.c, we can observe the outline (*i.e.* border) segmentation of characters “c”,“o” and “m”. Such cases of partial segmentation are not handled by the atom-based evaluator and counted as false positives, although the boundaries of these characters perfectly approximate their shapes and hence could successfully be recognized by an OCR. This example also illustrates a frequent problem involving characters with tittles (character “j”), which are missing from the segmentation result. The consequence is that the atom-based method considers the entire character “j” as not being segmented (false positive), decreasing both the recall (66.67%) and precision (75%).

In Fig. 2.d, one can see that the thickness of the segmentation of characters are slightly thicker than the GT. Here again the binary matching approach decision does not provide relevant comparisons between different segmentation cases. Due to the thickness variation, correct detections are counted as false positives and consequently both the recall and precision are under estimated (66.67% and 75%). This example also shows the difficulty of setting up correctly the decision thresholds and the huge impact of these parameters on the final scores.

Contrary to atom-based protocols, the pixel-based ones count all pixels that match the GT and reject all others. In the cases of Fig. 2.a, b and d, the recall seems to be more representative than that of the atom-based one. In Fig. 2.a, the four letters are well counted and the “filling” areas within the characters are considered as false positive pixels decreasing only the precision rate. The merged characters in Fig. 2.b are evaluate more fair than with the atom-based evaluation leading to a recall value of 92%. In Fig. 2.d, the thickness variation does not penalize the pixel-based recall (92.9%) as much as the atom-based one does (66.67). The large character area contributes however to an overestimation of recall (82.96%) and precision (93.75%) in Fig. 2.c. Here, the pixel evaluation method does not differentiate between the false positive pixels from character out-of-border pixels. Moreover, not all letters have the same weight, but highly depending on their area. For example, the characters in the word “life” contribute less to the overall recall than the characters of the word “jungle” because they have a bigger size. Similarly, because of the surface of the false positives in Fig. 2.a, the precision is severely underestimated (less then 50%).

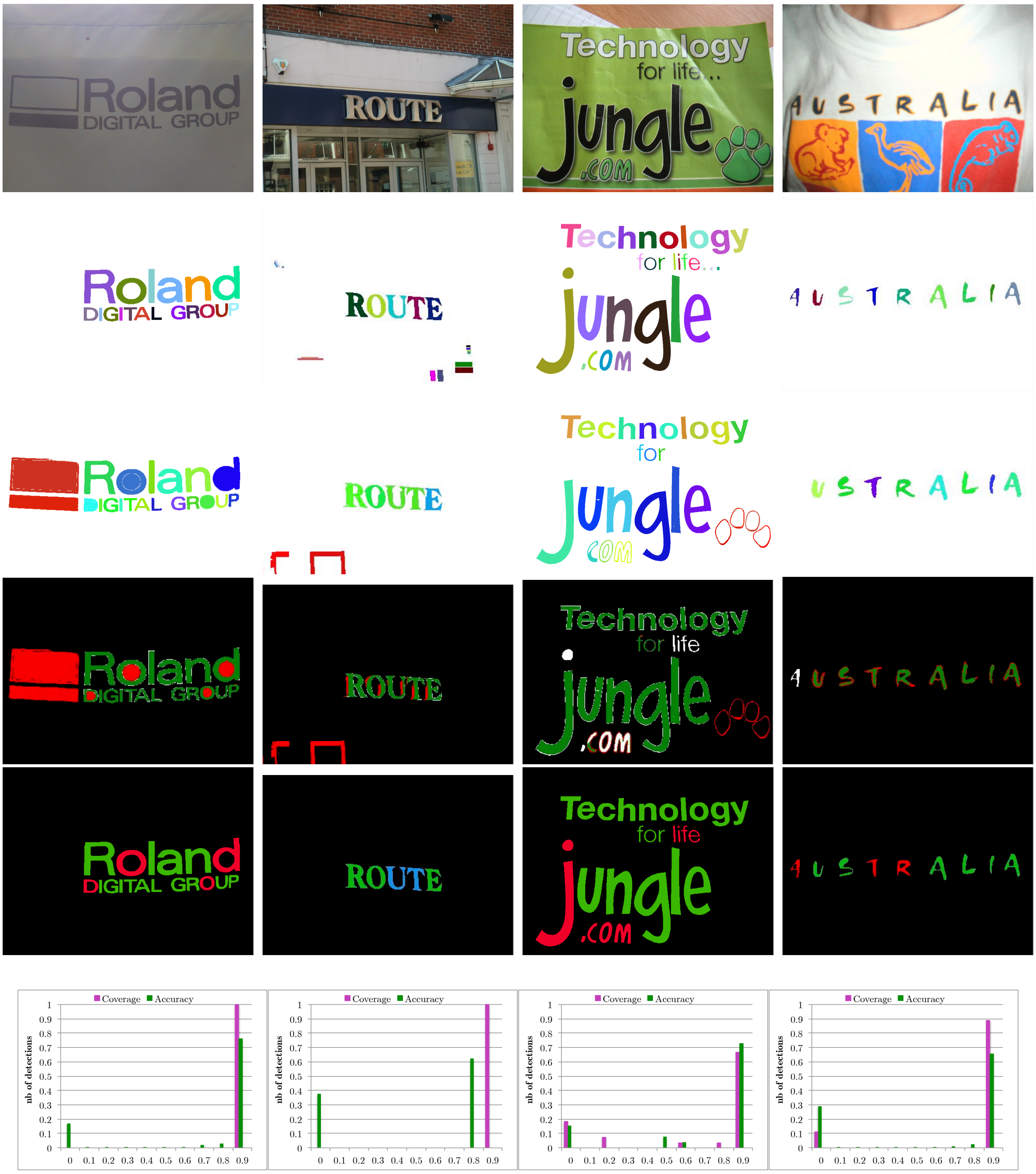
With our proposed method the filling of characters in Figure 2.a does not affect the recall, which correctly states that all characters were segmented (99.96%). The filling, however is penalized by the precision value. Similarly, the merged characters are correctly evaluated, producing an almost perfect recall score of 99.08%. The thickness variation that exceeds the allowed dilation of the character together with the three false positives lead to a precision score of 53%, which is rather logic if we assume that we have five out eight nearly correct segmentations. For the examples in Fig. 2.c the recall values are in between the ones produced by the atom-based and pixel evaluation protocols. This is consistent with the reasoning that the recall is over-estimated by the pixel evaluation due to the large text area and under-estimated by the atom-based one due to the minimal segmented area thresholds. The same is valid for the recall score obtained in Fig 2.d. In this picture, 8 characters are detected among the 9 characters. This leads to 88% of recall, which is the expected value correctly computed in spite of the thickness variation. The precision however seems over-estimated (79.24%) if compared to the rates obtained with pixel and atom based methods. This is due to two reasons: firstly, we count only one false positive ("A") instead of three as in the case of the atom-base evaluation, and secondly, by allowing the thickness variation we are more permissive than the pixel method.

4.2 Quantitative evaluation

Table 2 summarizes the rankings produced by the overall evaluation of nine text segmentation methods using four different protocols presented in Tables 3 and 4. As characterized by the standard deviation σ in Table 2 there is a significant discrepancy between some of the rankings produced by the pixel and atom-based evaluations (for example in the case of the *NSTextractor*). The difficulty now is to select the method that is the most reliable. To answer to this question, we have computed the mean of the ranking for each segmentation. First, one can observe that our method gives the nearest results to this mean. In a second time, we have ordered the segmentation results according to this mean. This gives us an average ranking. One can expect that, by collecting the rankings from various methods, the ranking would be smoothed and the impact of artifacts reduced in the evaluation protocols. Again our method is the nearest to this new ranking. This comparison to the mean ranking and to the new ranking is not an absolute prove but it provides however a reasonable clue for selecting the most appropriate evaluation method.

4.3 Results visualization

Fig. 3 illustrates the histogram representation [1] of the evaluation results of three segmentation methods (*NSTsegmentator* [19], *OCTYMIST* [17] and *StradVision* [24]) on the ICDAR 2015 natural scene dataset. This visualization tool is very useful as it provides at a glance important characteristics of the global segmentation efficiency, namely the distribution of coverage and accuracy values over an entire dataset. The first bin of the accuracy histogram represents the false positives, the last bin (*i.e.*]0.9,1]) represents the rate of perfect detections,



(a) (b) (c) (d)

Fig. 2: Four examples (a, b, c and d). For each example, from top to bottom: the original image, the GT, the segmentation, the pixel-based classification (false positive in red, correct in green, non-segmented in white), the atom-based classification (non-detected in red, correct in green, merged in blue) and the histogram visualization.

while all other bins represent the detection rates with accuracy values between 0.1 and 0.9. Similarly, the first bin of the coverage histogram represents the rate of GT objects not detected, while the last bin symbolizes the number of GT atoms that have been perfectly segmented. All other bins represent the GT atoms with coverage ratio between 0.1 and 0.9.

By analyzing the three histograms we can observe that the *StradVision* method achieves the highest accuracy peek, while keeping its false positive rate

Table 2: Ranking based on the F-Score and PERR scores in Tables 3 and 4. Mean and standard deviation of ranking. New ranking based on means of ranking.

	Ranking						
Participants	Pixel [20]	Atom [5]	Stathis [23]	Our's	Mean of ranking	σ	Ranking based on the means
BUCT_YST [11]	**3**	2	4	2	2.75	0.96	**3**
I2R_NUS [14]	**4**	6	3	**4**	4.25	1.26	**4**
I2R_NUS_FAR [14]	**2**	3	**2**	3	2.5	0.58	**2**
NSTextractor [19]	7	4	**6**	**6**	5.75	1,26	**6**
NSTsegmentator [19]	**8**	7	9	**8**	**8**	0.82	**8**
OTCYMIST [17]	**9**	**9**	8	**9**	8.75	0.5	**9**
StradVision [24]	**1**	**1**	**1**	**1**	**1**	0	**1**
TextDetection [7]	6	8	5	**7**	6,5	1.29	**7**
USTB_FuStar [29]	**5**	**5**	7	**5**	5.5	1	**5**

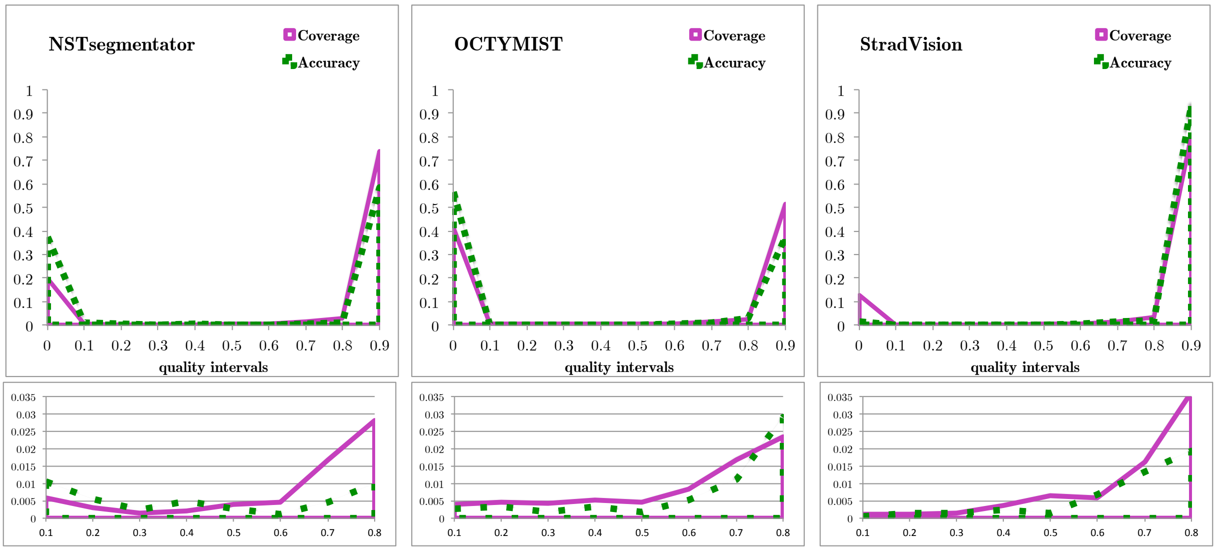
close to zero. On the contrary, one can observe an approximately 40% rate of false positives in the case of *NSTsegmentator* method, and a significantly larger rate for the *OCTYMIST* method, around 60%. Concerning the coverage values, one can observe a narrower difference between the segmentation misses of *NSTsegmentator* and that of *StradVision*, both around 70%. Another interesting aspect of this analysis consists in the fair distribution of missed and perfect atom segmentations produced by the *OCTYMIST* method (the value in the first and last bin are very close). By looking at the three histograms we can identify the same pattern, where most of the values fall in the first and last bins, meaning that all GT atoms where either well segmented or not at all, or that all produced segmentations perfectly match the GT atoms or they completely missed them. However, distributions of the accuracy and coverage values also appear within the bins between 0.1 and 0.9 as illustrated in the bottom of Fig. 3, although in a less important number.

One of the advantage of unifying the localization evaluation of EvaLTex with the segmentation evaluation is to benefit from this visualization scheme. To our knowledge, such a visualization tool for analyzing the precision of segmentation results does not yet exist. This visualization scheme can also be apply independently on segmentation examples as seen in Fig 2.

5 Conclusion

In this article, we have presented a framework to evaluate text segmentation results. This framework was inspired from EvaLTex, a protocol initially designed to evaluate text detection systems.

The proposed evaluation scheme is able to deal efficiently with different possible segmentation scenarios such as broken, merged characters, partially detected characters or both broken and merged characters. Its robustness to slight variations in character thickness makes the protocol independent of the character's



(a) (b) (c)

Fig. 3: Histogram representation of the coverage and accuracy rates on the ICDAR 2015 natural scene dataset of three segmentation methods: (a) *NSTsegmentator* [19]; (b) *OCTYMIST* [17]; (c) *StradVision* [24]. The bottom histograms represent the detailed distribution of rates over the intervals between 0.1 and 0.8. One can observe that the histograms provide a good inside on the behavior of a text segmentation algorithm. For example, the most powerful method is *StradVision* as it has the highest accuracy and coverage values in the last bin of the histogram. On the other hand, *OCTYMIST* has a higher rate of false positives (green values in the first bin) than correct character detections (green value in the last bin). Also, the number of true positives (magenta value in the last bin) is equal to the number of missed detection (magenta value in the first bin). The *NSTsegmentator* lies somewhere in between the two previous methods as it has a lower number of missed detections and false positives than *OCTYMIST* but higher than *StradVision*.

size. Moreover, the evaluation does not rely on a binary decision, which allows a better discrimination between different segmentation scenarios. For all these reasons, scores computed using this framework are not only representative for the segmentation quality but they also allow to characterize and compare segmentation methods. To sum up, as this work handles the segmentation evaluation (at character level) and is an adaptation of a framework capable of evaluating text detection at paragraph/line or word levels, we can state that the presented work is a unified tool to evaluate a text understanding chain at every stage of the detection process. Furthermore, we have shown that we can successfully apply the histogram visualization tool to segmentation evaluation results in order to provide a more detailed overview of the segmentation method's behavior.

In the future, a possible improvement would be to add an OCR evaluation step in the EvaLTex framework. This platform is already able the provide the correspondence between objects in the detection and objects in the GT. This would indicate which character in the GT would be associated to which character in the detection set. The release of this tool is equally planned.

Table 3: Quality, quantity and global scores obtained with alternative evaluation methods (Pixel, atom and Stiathos's method) on the detection results of all participants (in ascending order) during ICDAR'13 and ICDAR'15 Robust Reading Competition (Challenge 2, Task 2).

	Pixel evaluation [20]			Atom evaluation [5]			Stathis [23] evaluation			
Participants	Recall	Precision	F-Score	Recall	Precision	F-Score	*PERR*	*MSE*	*SNR*	*PSNR*
BUCT_YST [11]	75.56	81.75	77.99	71.13	83.51	76.83	0.0235312	1530.12	39.8059	79.2717
I2R_NUS [14]	73.57	79.04	76.21	60.33	76.62	67.51	0.0233201	1516.39	39.3688	78.8345
I2R_NUS_FAR [14]	74.73	81.70	78.06	68.64	80.59	74.14	0.0228714	1487.22	39.73	79.1958
NSTextractor [19]	60.71	76.28	67.61	63.38	83.57	72.09	0.0306271	1991.53	38.2217	77.6875
NSTsegmentator [19]	68.41	63.95	66.10	68.00	54.35	60.41	0.0433068	2816.02	36.2739	75.7397
OTCYMIST [17]	46.11	58.53	51.58	41.79	31.60	35.99	0.0415516	2701.9	36.1228	75.5886
StradVision [24]	78.80	89.24	83.70	73.02	84.42	78.31	0.0187627	1220.04	40.512	79.9777
TextDetection [7]	64.74	76.20	70.01	62.03	57.43	59.64	0.0297574	1934.98	38.621	78.0868
USTB_FuStar [29]	69.58	74.45	71.93	68.03	72.46	70.18	0.0310387	2018.29	38.6495	798.1152

Table 4: Quality, quantity and global scores obtained with our evaluation method on the detection results of all participants (in ascending order) during ICDAR'13 and ICDAR'15 Robust Reading Competition (Challenge 2, Task 2).

	Quality		Quantity		Global		
Participants	Recall	Precision	Recall	Precision	Recall	Precision	F-Score
BUCT_YST [11]	97.73	97.81	84.25	95.92	82.34	93.83	87.71
I2R_NUS [14]	96.94	96.91	81.17	93.06	78.68	90.18	84.04
I2R_NUS_FAR [14]	96.92	97.09	82.22	93.14	79.69	90.43	84.72
NSTsegmentator [19]	96.59	98.46	70.66	92.31	68.26	90.89	77.96
NSTsegmentator [19]	95.91	95.45	80.96	63.04	77.66	60.17	67.81
OCTYMIST [17]	93.23	94.73	59.76	43.63	55.72	41.33	47.46
StradVision [24]	97.36	98.32	87.52	98.34	85.21	96.69	90.59
TextDetection [7]	95.83	97.24	79.27	71.22	75.97	69.25	72.46
USTB_FuStar [29]	98.01	97.31	80.59	83.10	78.99	80.87	79.92

References

1. Calarasanu, S., Fabrizio, J., Dubuisson, S.: Using histogram representation and earth mover's distance as an evaluation tool for text detection. In: Proc. International Conference on Document Analysis and Recognition (2015)
2. Calarasanu, S., Fabrizio, J., Dubuisson, S.: What is a good evaluation protocol for text localization systems? concerns, arguments, comparisons and solutions. Image and Vision Computing 46, 1 – 17 (2016), `http://www.sciencedirect.com/science/article/pii/S0262885615001377`
3. Cao, H., Govindaraju, V.: Handwritten carbon form preprocessing based on markov random field. In: Proc. Computer Vision and Pattern Recognition. pp. 1–7 (June 2007)
4. Chang, F., Liang, K.H., Tan, T.M., Hwan, W.L.: Binarization of document images using hadamard multiresolution analysis. In: Proc. International Conference on Document Analysis and Recognition. pp. 157–160 (1999)
5. Clavelli, A., Karatzas, D., Llados, J.: A framework for the assessment of text extraction algorithms on complex color images. In: Proc. Document Analysis Systems. pp. 19–26 (2010)
6. Fabrizio, J., Marcotegui, B., Cord, M.: Text segmentation in natural scenes using toggle-mapping. In: Proceedings of the 16th IEEE International Conference on Image Processing. pp. 2349–2352 (2009)
7. Fabrizio, J., Robert-Seidowsky, M., Dubuisson, S., Calarasanu, S., Boissel, R.: Textcatcher: a method to detect curved and challenging text in natural scenes. International Journal on Document Analysis and Recognition 19(2), 99–117 (2016)
8. Filho, C.J.A.B., Mello, C.A.B., Andrade, J.D., Falcao, D.M.A., Lima, M.P., Santos, W.P., Oliveira, A.L.I.: Based on color quantization by genetic algorithms. In: 19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007). vol. 1, pp. 488–491 (Oct 2007)
9. Galibert, O., Kahn, J., Oparin, I.: The zonemap metric for page segmentation and area classification in scanned documents. In: Proc. International Conference on Image Processing. pp. 2594–2598 (2014)
10. He, J., Do, Q.D.M., Downton, A.C., Kim, J.H.: A comparison of binarization methods for historical archive documents. In: Proc. International Conference on Document Analysis and Recognition. pp. 538–542 (Aug 2005)
11. Huang, W.: Buctyst segmentation method. `http://rrc.cvc.uab.es/?com=evaluation&ch=2&view=task2_method&id_submit=2608`
12. Kang, B.H., Han, G.S., Kim, H.G., Kim, J.S., Yoon, C.R., Cho, M.S.: Fuzzy inference and logical level methods for binary graphic/character image extraction. In: Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on. vol. 5, pp. 4626–4629 (Oct 1998)
13. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: Proc. International Conference on Document Analysis and Recognition (2015)
14. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., de las Heras, L.P.: ICDAR 2013 robust reading competition. In: Proc. International Conference on Document Analysis and Recognition. pp. 1484–1493 (2013)
15. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., Bigorda, L.G.i., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., de las Heras, L.P.: Icdar 2013 robust reading

competition. In: International Journal on Document Analysis and Recognition. pp. 1484–1493 (2013)

16. Kohmura, H., Wakahara, T.: Determining optimal filters for binarization of degraded characters in color using genetic algorithms. In: Proc. International Conference on Pattern Recognition. vol. 3, pp. 661–664 (2006)
17. Kumar, D., Ramakrishnan, A.G.: Otcymist: Otsu-canny minimal spanning tree for born-digital images. In: Proc. Document Analysis Systems. pp. 389–393 (March 2012)
18. Mao, S., Kanungo, T.: Software architecture of pset: A page segmentation evaluation toolkit. International Journal on Document Analysis and Recognition 4(3), 205–217 (2002)
19. Milyaev, S., Barinova, O., Novikova, T., Kohli, P., Lempitsky, V.: Image binarization for end-to-end text understanding in natural images. In: Proc. International Conference on Document Analysis and Recognition. pp. 128–132 (2013)
20. Ntirogiannis, K., Gatos, B., Pratikakis, I.: An objective evaluation methodology for document image binarization techniques. In: Proc. Document Analysis Systems. pp. 217–224 (Sept 2008)
21. Robert-Seidowsky, M., Fabrizio, J., Dubuisson, S.: TextTrail: a robust text tracking algorithm in wild environments. In: Proceedings of the 10th International Conference on Computer Vision Theory and Applications. pp. 268–276 (Mar 2015)
22. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging 13(1), 146–168 (2004), `http://dblp.uni-trier.de/db/journals/jei/jei13.html#SezginS04`
23. Stathis, P., Kavallieratou, E., Papamarkos, N.: An evaluation technique for binarization algorithms. Journal of Universal Computer Science 14(18), 3011–3030 (oct 2008)
24. Sung, M.C., Jun, B., Cho, H., Kim, D.: Scene text detection with robust character candidate extraction method. In: Proc. International Conference on Document Analysis and Recognition. pp. 426–430 (Aug 2015)
25. Trier, O.D., Taxt, T.: Evaluation of binarization methods for document images. Pattern Analysis and Machine Intelligence 17(3), 312–315 (Mar 1995)
26. Wang, Q., Tan, C.L.: Matching of double-sided document images to remove interference. In: Proc. Computer Vision and Pattern Recognition. vol. 1, pp. I–1084–I–1089 vol.1 (2001)
27. Wolf, C., Jolion, J.M., Chassaing, F.: Text localization, enhancement and binarization in multimedia documents. In: Proc. International Conference on Pattern Recognition. vol. 2, pp. 1037–1040 (2002)
28. Wolf, C., Jolion, J.M.: 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)
29. Yin, X.C., Yin, X., Huang, K., Hao, H.W.: Robust text detection in natural scene images. Pattern Analysis and Machine Intelligence 36(5), 970–983 (May 2014)
30. Zhu, Y., Wang, C., Dai, R.: Document image binarization based on stroke enhancement. In: Proc. International Conference on Pattern Recognition. vol. 1, pp. 955–958 (2006)