READ-BAD: A New Dataset and Evaluation Scheme for Baseline Detection in Archival Documents

Tobias Grüning, Roger Labahn
Computational Intelligence Technology Lab
University of Rostock
18057 Rostock, Germany
{tobias.gruening, roger.labahn}@uni-rostock.de

Markus Diem, Florian Kleber and Stefan Fiel
Computer Vision Lab
TU Wien
1040 Vienna, Austria
{diem,kleber,fiel}@caa.tuwien.ac.at

Abstract—Text line detection is crucial for any application associated with Automatic Text Recognition or Keyword Spotting. Modern algorithms perform good on well-established datasets since they either comprise clean data or simple/homogeneous page layouts. We have collected and annotated 2036 archival document images from different locations and time periods. The dataset contains varying page layouts and degradations that challenge text line segmentation methods. Well established text line segmentation evaluation schemes such as the *Detection Rate* or *Recognition Accuracy* demand for binarized data that is annotated on a pixel level. Producing groundtruth by these means is laborious and not needed to determine a method's quality. In this paper we propose a new evaluation scheme that is based on baselines. The proposed scheme has no need for binarization, it can handle skewed and rotated text lines and its results correlate with Handwritten Text Recognition accuracy. The ICDAR 2017 Competition on Baseline Detection and the ICDAR 2017 Competition on Layout Analysis for Challenging Medieval Manuscripts make use of this evaluation scheme.

I. INTRODUCTION

Layout analysis (LA) is considered an open research topic especially for historical collections in the document analysis community and is a major pre-processing step for e.g. Keyword Spotting (KWS) or Handwritten Text Recognition (HTR). In the last years several competitions were organized to evaluate the performance of layout analysis algorithms: Some focusing purely on LA [1]–[6], some requiring a good LA as pre-processing step to achieve competitive results [7]–[9]. The ongoing effort in organizing such competitions strongly indicates that there is still a need for improvement concerning LA.

Even state-of-the-art algorithms have problems if they are faced with degradations related to historical documents [6], e.g. faded-out ink, bleed-through, marginalia, skewed and touching/overlapping text lines. In contrast, reported results of LA algorithms perform surprisingly well with accuracies far better than 90% [10]–[17]. This is basically due to the fact that the well established easily accessible datasets (like the IAM-HistDB consisting of *Saint Gall Database* [18], *Parzival Database* [19] and *Washington Database* [19], as well as the datasets provided via the competitions [1], [3], [5], the datasets

introduced in [14] and even newly proposed datasets like the collection of Southeast Asian palm leaf manuscript images [20]) are not covering the full range of difficulties present in historical documents. The datasets contain either modern, well aligned handwritten texts without any serious difficulties for state-of-the-art algorithms at all or very homogeneous layouts within a dataset, hence it is an ease to adapt algorithms to such datasets to achieve high accuracy.

Since state-of-the-art methods achieve high accuracies on well-established datasets, there is a need for a newly annotated dataset with more challenging page layout and a greater variety in terms of script, time range and place of origin. A huge variety of degradation as well as different resolutions and orientations should be present. Since the landscape of document analysis has changed over the last years, and machine learning based algorithms get more and more popular not only for KWS [21] and HTR [22] but also for LA [23]–[25], the dataset should consist of hundreds of pages to provide an appropriate amount of training samples.

Besides the characteristics of the images the kind of groundtruth (GT) provided is essential. The variety of GT given for different datasets ranges from origin points [6] over polygons surrounding the text lines [18], [19] and groundtruth on pixel level [1], [3], [14] to detailed information about text region entities [4] and reading order [8]. Since in the most application scenarios LA is mainly a pre-processing step for HTR, it is meaningful to provide goal-oriented GT. Modern HTR systems require text lines as inputs [21], [22], that is why we will restrict ourselves to the text line detection scenario and ignore issues like entity classification and reading order. Nevertheless in complex layout scenarios (e.g. tables, multi-column texts, present marginalia), it is mandatory to detect the page layout to achieve correct text line segmentation results. Ignoring the page layout typically leads to an undersegmentation of text lines. Therefore, the text line segmentation scenario somehow comprises the page segmentation scenario as a required intermediate processing step.

To characterize the text lines using solely origin points is in our opinion not sufficient since they don't cover the character-

Fig. 1. Illustration of a baseline.

istics, e.g. skew, orientation, dimension, ... , of the text lines at all. On the other hand, [26] showed that the HTR accuracy is not significantly effected by the polygon surrounding the text lines. Even simple strategies to construct surrounding polygons given baseline representations lead to satisfying results [26]. Therefore, GT based on baseline representations for the text lines is in our opinion a reasonable compromise. Furthermore, annotating baselines is less cumbersome than surrounding polygons and therefore cheaper.

Since the widely-used evaluation schemes rely on surrounding polygons and use area (or foreground pixel) based methods to calculate the accuracy of text line segmentation results, there is a need for an evaluation scheme suitable for baselines.

In this paper, we introduce a new dataset containing 2036 pages of historical documents with annotated baselines. Furthermore, we propose a newly developed, goal-oriented evaluation scheme working with baseline representations of the text lines. The remaining paper is structured as follows, in Section II the dataset is described in detail, a meaningful subdivision is explained and some example pages as well as statistics are shown. Section III describes the newly proposed evaluation scheme along with some examples demonstrating the functionality of the scheme. Section IV concludes the paper.

II. DATASET

The ICDAR 2017 Competition on Baseline Detection (cBAD) dataset [27] consists of 2036 document page images that were collected from 9 different archives. It is to the best of our knowledge the first text line segmentation dataset that relies on baselines only.

A. Baseline Definition

A baseline is defined in the typographical sense as the virtual line where most characters rest upon and descenders extend below (see Figure 1). Any text line that contains textual information is annotated by one single baseline. Hence, non-textual symbols (including decorations lines, dotted lines, images, noise/stains, initials, bleed-through text) are not annotated. Curved text lines are approximated by a baseline using multiple points. Baselines are split if

- they span different columns (see Figure 2).
- they span different document pages (see Figure 2).
- between marginalia and the body text (see Figure 2 top).

If a text line is clearly not part of a table (column) system, a single baseline is annotated (see Figure 2).
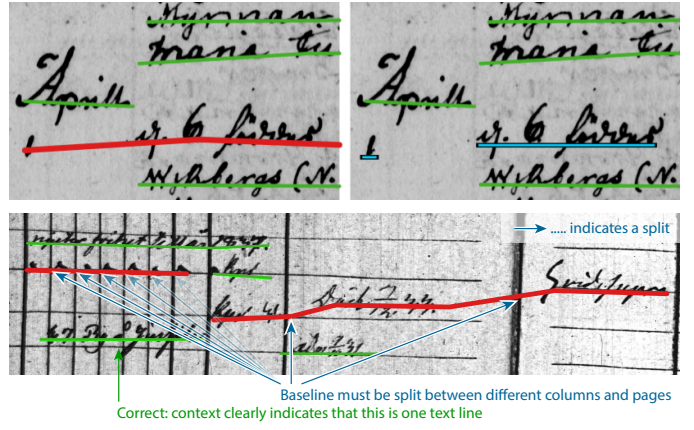

Fig. 2. Example of complex text lines where red (bold) baselines indicate wrongly annotated text lines.

B. The cBAD Dataset

About 2000 document images written between 1470 and 1930 were collected from 9 different European archives. We sampled 250 images from each archival collection using a freely available python script[1]. A more detailed description of the document collections is given below.

Archive Bistum Passau (ABP): collection contains 16,000 images photographed at 300 dpi. The documents include parish registers of baptisms, marriages, and funerals.

Bohisto - Bozen State Archive: 77,000 page images of council minutes written between 1470 and 1804.

Venice Time Machine (EPFL): about 5000 pages from indexes of records, records of real property transactions, and daily death registrations written between the 16th and 18th century.

Humboldt University Berlin (HUB): 3600 student notes of lectures given by Alexander von Humboldt between 1827 and 1829.

National Archive Finland (NAF): 2186 page images from account books, a court book, a census book, and a church book that cover a time period from 1774 until the 1930s.

Marburg State Archive: 36,000 page images from the Grimm collection comprising letters, postcards, and greeting cards.

University College London (UCL): the Bentham papers include 55,000 pages. Most pages were written by the British philosopher Jeremy Bentham between 1760 and 1832.

C. Data Annotation

In total 2250 images were collected. Before groundtruthing we removed 132 images due to poor quality and content (e.g. music scores). The 2118 remaining images were annotated by DigiTexx. Afterwards, the GT was inspected by two independent operators who removed another 82 images because of wrong baseline annotations resulting in a final dataset size of 2036 images.

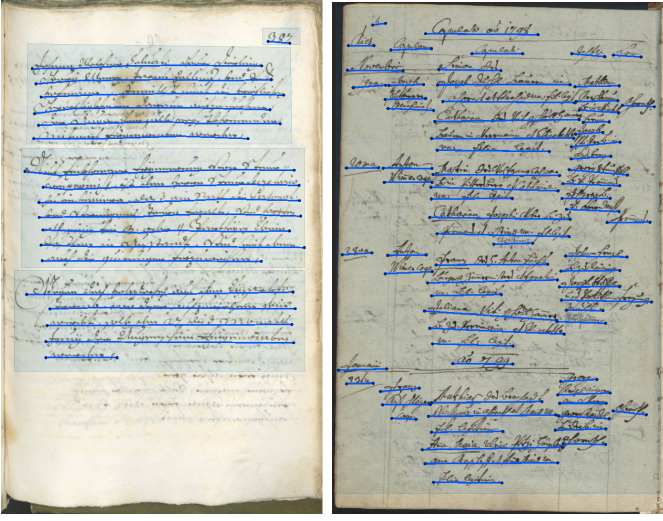[1]https://github.com/TUWien/Benchmarking

Fig. 3. Two examples of document images of TRACK A Simple Documents (left) and TRACK B Complex Documents (right) with annotated baselines and text regions.

The annotated dataset is split into two tracks: *Simple Documents* and *Complex Documents*. The former includes only pages with simple page layouts and annotated text regions. Hence, this track is used to evaluate the text line segmentation only, thus neglecting issues that arise from the page layout. The second track *Complex Documents* includes full page tables, multi column text and rotated text lines. The challenge is not only to robustly detect baselines but also to split baselines correctly with respect to the page layout.

Since there are supervised baseline detection methods, we split both tracks into a training and a test set. For training 30 images are taken from each collection resulting in 216 training images for *Simple Documents* and 270 images for *Complex Documents*. The data along with the GT is publicly available [27] for both training sets. For the test sets, the GT will be released after the competition deadline.

The well-known PAGE XML[2] scheme is used for storing text regions and baselines.

III. EVALUATION SCHEME

Since baseline detection is basically a pre-processing step for HTR, there are special requirements regarding the evaluation scheme:

- Results should correlate with HTR accuracy (there is not an unique correct baseline, slightly different baselines lead to the same HTR accuracy)
- The evaluation scheme should reflect how much of the text – ignoring layout issues – was detected (we call the value reflecting this R value, since it has similar properties as the well established *recall* value)
- The evaluation scheme should reflect how reliable the structure of the text lines (the layout) of the document was detected (we call the value reflecting this P value, since

it has similar properties as the well established *precision* value)
- The evaluation scheme should not rely on binarization, because there are various algorithms explicitly avoiding binarization [12], [15], [24]
- The evaluation scheme should be able to handle skewed and oriented text lines
- The evaluation scheme should not rely on a reading order

To our knowledge there is no well-established evaluation scheme meeting these requirements. Hence, we propose a newly developed scheme to evaluate the performance of baseline detection algorithms. The proposed algorithm is implemented in java and available as a standalone command line tool, which licensed as LGPLv3 and publicly available[3].

A. Single Page Evaluation

In the following the calculation of R and P for a single page is explained. Let $\mathcal{P}$ be the set of all polygonal chains (each polygonal chain represents a baseline and contains a finite number of vertices characterized by two coordinates). $\mathcal{G} = \{\boldsymbol{g}_1, ..., \boldsymbol{g}_M\} \subset \mathcal{P}$ is the set of given (GT) polygonal chains representing the baselines for a single page and $\mathcal{H} = \{\boldsymbol{h}_1, ..., \boldsymbol{h}_K\} \subset \mathcal{P}$ is the set of hypothesis (HY) polygonal chains calculated by a baseline detection algorithm for the same page, Fig. 4a. The following describes the calculation of R and P for the two sets $\mathcal{G}$ and $\mathcal{H}$.

1) Polygonal Chain Normalization: In a first step each chain is normalized, so that two adjacent vertices are in the 8-neighborhood of each other (have a distance $\leq \sqrt{2}$), Fig. 4b. The resulting sets of normalized chains are $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{P}}$. For better readability we omit the tilde. In the following $\mathcal{G}$ and $\mathcal{P}$ are the sets of normalized polygonal chains.

2) Tolerance Value Calculation: In a second step for each chain $\boldsymbol{g} \in \mathcal{G}$ a *tolerance value* $t_{\boldsymbol{g}}$ is calculated. As mentioned above, the evaluation scheme should not penalize HY baselines which are slightly different to the GT baselines. Hence, some kind of tolerance is necessary. Page (and text line) dependent tolerance values are calculated, because within a collection various resolutions and layout scenarios could be present. A single pre-defined tolerance value can hardly cover all these scenarios in a satisfying fashion. Since the y-coordinates of the vertices are typically "wrongly" oriented in computer vision scenarios they have to be negated for the following procedure. To calculate $t_{\boldsymbol{g}}$, the orientation $\alpha_{\boldsymbol{g}} \in [0, \pi)$ of $\boldsymbol{g}$ is estimated using linear regression. $\boldsymbol{o}(\alpha_{\boldsymbol{g}}) = (\cos(\alpha_{\boldsymbol{g}}), \sin(\alpha_{\boldsymbol{g}}))^T$ is the vector of length 1 of orientation $\alpha_{\boldsymbol{g}}$. Given the set $\mathcal{V}$ of all vertices of the chains in $\mathcal{G} \setminus \{\boldsymbol{g}\}$, the subset $\mathcal{V}_{\boldsymbol{g}} \subset \mathcal{V}$ is calculated such that for any $\boldsymbol{v} \in \mathcal{V}_{\boldsymbol{g}}$ there are at least two vertices $\boldsymbol{v}_1, \boldsymbol{v}_2 \in \boldsymbol{g}$ satisfying

$$(\boldsymbol{v} - \boldsymbol{v}_1)^T \boldsymbol{o}(\alpha_{\boldsymbol{g}}) \cdot (\boldsymbol{v} - \boldsymbol{v}_2)^T \boldsymbol{o}(\alpha_{\boldsymbol{g}}) \leq 0. \quad (1)$$

Condition (1) means that the projections of $(\boldsymbol{v} - \boldsymbol{v}_1)$ and $(\boldsymbol{v} - \boldsymbol{v}_2)$ into the direction of $\boldsymbol{o}(\alpha_{\boldsymbol{g}})$ have different algebraic signs (or have length zero). In Fig. 4c the set $\mathcal{V}_{\boldsymbol{g}_2}$ of vertices for
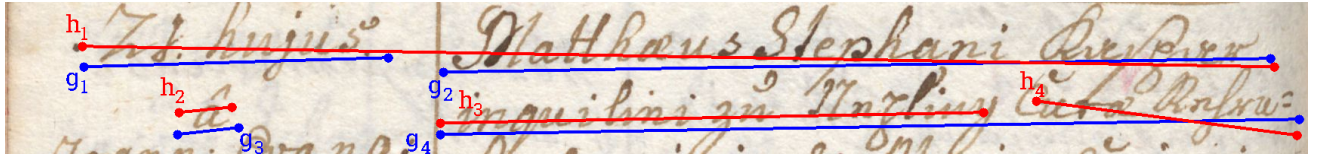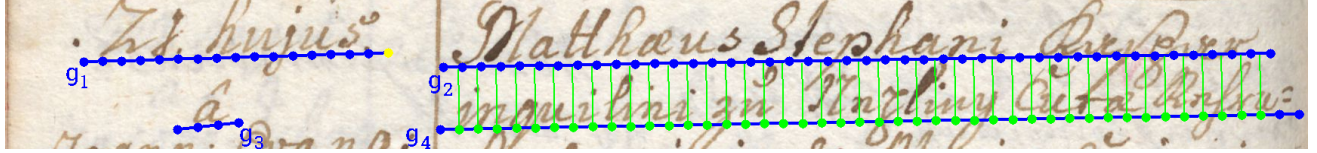
(a) Depicted are the sets $\mathcal{G} = \{g_1, g_2, g_3, g_4\}$ of four GT baselines (blue) and $\mathcal{H} = \{h_1, h_2, h_3, h_4\}$ of four HY baseline (red).
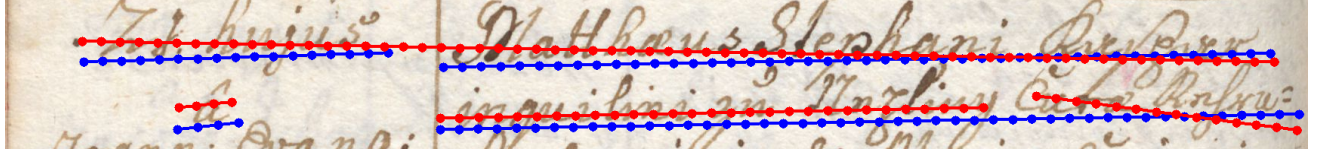
(b) The same baselines as shown in Fig. 4a but represented by normalized polygonal chains (for better clarity only every 25th vertex is shown).

(c) The vertices, which are taken into account to calculate the minimum distance of GT line 2 to the other GT lines, are displayed as green points. Green lines are the orthogonal (to GT line 2) distances of the green vertices.
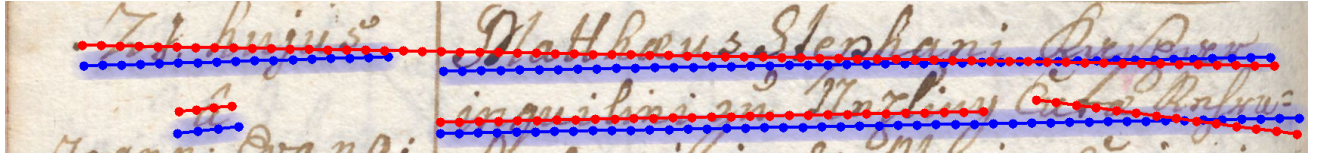
(d) Shown in light blue are the tolerance areas for the different GT baselines, for all four baselines the estimated tolerance value is roughly 20.

Fig. 4. Depicted is a snippet of an example document and intermediate steps of the evaluation scheme.

GT baseline 2 is shown (green points). For each $v \in \mathcal{V}_g$ one vertex $v_m(v) \in g$ is determined for which the projection of $(v - v_m(v))$ into the direction of $o(\alpha_g)$ has minimal length

$$v_m(v) = \arg\min_{v_g \in g} \left| (v - v_g)^T o(\alpha_g) \right|.$$

The minimum distance of g to another chain is calculated by

$$d_g = \min_{v \in \mathcal{V}_g} \left| (v - v_m(v))_x o(\alpha_g)_y - (v - v_m(v))_y o(\alpha_g)_x \right|.$$

Subscripts v_x and v_y are the x- and y-coordinate of vector v. d_g is the minimal length of the projections of all $(v - v_m(v))$ into the direction orthogonal to $o(\alpha_g)$, see Fig. 4c (green lines). For $\mathcal{V}_g = \emptyset$ there are no other baselines allowing a meaningful calculation of d_g, hence it is set to some default value (250 was chosen). Condition (1) is essential since $\mathcal{V}_g$ is the basis for the estimation of the minimal distance of g to another chain. For instance the yellow vertex Fig. 4c has a significantly shorter orthogonal projection to GT line 2, but of course would falsify the statistics. The mean $\overline{d}_\mathcal{G}$ over the d_g ($g \in \mathcal{G}$) with a value different to the default value is calculated. Finally, the GT baseline dependent tolerance values are calculated, facilitating

$$t_g = 0.25 \cdot \min(d_g, \overline{d}_\mathcal{G}).$$

25% of the estimated interline distance yields a reasonable compromise between accuracy and flexibility. $\mathcal{T} = \mathcal{T}(\mathcal{G})$ is

Algorithm 1 Coverage Function

1: **procedure** COV(p, q, t)
2:     $c \leftarrow 0$
3:     **for** $p = (p_x, p_y)$ vertex of p **do**
4:         $d_{min} \leftarrow \min_{q \in q}(\|p - q\|_2)$
5:         **if** $d_{min} \leq t$ **then**
6:             $c \leftarrow c + 1$
7:         **else if** $d_{min} \leq 3t$ **then**
8:             $c \leftarrow c + \frac{3t - d_{min}}{2t}$
9:         **end if**
10:     **end for**
11:     $c \leftarrow \frac{c}{|p|}$ ▷ $|p|$ is the number of vertices of p
12:     **return** c
13: **end procedure**

the set containing the resulting tolerance values, in Fig. 4d the blue areas show the individual tolerance areas for the different GT baselines.

3) Coverage Function: Employing the (tolerance dependent) COV : $\mathcal{P} \times \mathcal{P} \times \mathbb{R} \to \mathbb{R}$ function implemented via Alg. 1, one can determine a value representing the fraction of chain p for which there is a vertex of chain q within a certain tolerance area (skew-invariant). Alg. 1 counts the number of vertices of p for which there is a vertex of q with a distance less than the given tolerance value t. Furthermore a smooth

(linear) transition is performed for vertices with a distance between t and $3t$. A vertex with a distance less than t counts 1, with a distance of $1.5t$ it counts 0.75, with a distance of $2t$ it counts 0.5, ... Finally, a vertex with a distance of $3t$ and more counts 0. The resulting value is normalized using the number of vertices of $\boldsymbol{p}$.

Let $\text{COV}_S : \mathcal{P} \times \mathfrak{P}(\mathcal{P}) \times \mathbb{R} \to \mathbb{R}$ be the generic extension of COV to sets of polygonal chains as second argument. The minimum from line 4 in Alg. 1 is calculated over a set of chains instead of a single chain. To clarify the functionality of the coverage functions a few exemplary values are shown in Tab. I. Especially, the function COV is not commutative in the first two arguments.

TABLE I
EXAMPLE VALUES OF THE COVERAGE FUNCTIONS APPLIED TO NORMALIZED POLYGONAL CHAINS SHOWN IN FIG. 4B WITH A FIXED TOLERANCE VALUE OF 20 (AS SHOWN IN FIG. 4D). g_i MEANS THE NORMALIZED VERSION OF THE i-TH GT BASELINE.

$\boldsymbol{p}$	$\boldsymbol{q}$	$\boldsymbol{r}$	$\text{COV}(\boldsymbol{p},\boldsymbol{q},20)$	$\text{COV}_S(\boldsymbol{p},\{\boldsymbol{q},\boldsymbol{r}\},20)$
h_3	g_4	–	1.0	–
g_4	h_3	h_4	0.65	0.96
g_3	h_2	–	0.76	–
h_1	g_1	g_2	0.26	0.95

4) R and P Calculation: The tolerance dependent R value of $\mathcal{G}$ and $\mathcal{H}$ is finally calculated by

$$\text{R}(\mathcal{G}, \mathcal{H}, \mathcal{T}) = \frac{\sum_{\boldsymbol{g} \in \mathcal{G}} \text{COV}_S(\boldsymbol{g}, \mathcal{H}, t_{\boldsymbol{g}})}{|\mathcal{G}|}. \tag{2}$$

The R value indicates for what fraction of the GT baselines there are detected HY baselines within a certain tolerance area. Segmentation (page layout) errors are not penalized at all, because no alignment between GT and HY baselines is enforced.

These segmentation errors are penalized in the P value. Let $\mathcal{M}(\mathcal{G}, \mathcal{H}) \subset \mathcal{G} \times \mathcal{H}$ be an alignment of GT and HY chains where each element of $\mathcal{G}$ as well as of $\mathcal{H}$ occurs at most once. The tolerance dependent P value of $\mathcal{G}$ and $\mathcal{H}$ is calculated as follows

$$\text{P}(\mathcal{G}, \mathcal{H}, \mathcal{T}) = \frac{\sum_{(\boldsymbol{g},\boldsymbol{h}) \in \mathcal{M}(\mathcal{G},\mathcal{H})} \text{COV}(\boldsymbol{h}, \boldsymbol{g}, t_{\boldsymbol{g}})}{|\mathcal{H}|}. \tag{3}$$

An alignment ensures that segmentation errors are penalized. E.g. if a text line is splitted into two equally sized parts, a R value of 1.0 is calculated (the two detected chains cover the entire GT chain), but the expected P value is 0.5 (the GT chain is aligned with exactly one of the HY chains with a P value of 1, this is divided by 2, because there are two HY chains). We want to mention that for both cases (R and P) short text lines have the same impact as long ones, because in (2) and (3) the line specific R and P values are divided by the number of GT respectively HY lines. This prevents the proposed evaluation scheme from underestimating the importance of short text lines, which often contain essential information in the context of historical documents, e.g. dates.

5) Greedy-based Alignment: To evaluate (3) an P-optimal alignment is necessary. Therefore a P matrix $\boldsymbol{C} \in \mathbb{R}^{M \times K}$ is calculated with elements $c_{ij} = \text{COV}(\boldsymbol{h}_i, \boldsymbol{g}_j, t_{\boldsymbol{g}_j})$. Based on this, the alignment is calculated in a greedy manner $\mathcal{M}(\mathcal{G}, \mathcal{H}) = \text{ALIGN}(\boldsymbol{C}, \mathcal{G}, \mathcal{H})$, see Alg. 2. A greedy approach was chosen, because there is no reading order available (no dynamic programming possible) and the greedy solution is in most practical cases the exact solution.

Algorithm 2 Alignment Function

1: **procedure** ALIGN($\boldsymbol{C}, \mathcal{G}, \mathcal{H}$)
2: $\quad \mathcal{M} \leftarrow \emptyset$
3: $\quad \boldsymbol{C}' \leftarrow \boldsymbol{C}$
4: $\quad$ **while** $\boldsymbol{C}'$ is not empty **do**
5: $\quad\quad m \leftarrow$ one of the maximal elements of $\boldsymbol{C}'$
6: $\quad\quad$ **if** $m > 0$ **then**
7: $\quad\quad\quad$ //create a new matching pair
8: $\quad\quad\quad \boldsymbol{g} \leftarrow$ element of $\mathcal{G}$ belonging to m
9: $\quad\quad\quad \boldsymbol{h} \leftarrow$ element of $\mathcal{H}$ belonging to m
10: $\quad\quad\quad \mathcal{M} \leftarrow \mathcal{M} \cup (\boldsymbol{g}, \boldsymbol{h})$
11: $\quad\quad\quad \boldsymbol{C}' \leftarrow$ take $\boldsymbol{C}'$ and delete row/col of m
12: $\quad\quad$ **else**
13: $\quad\quad\quad$ **return** $\mathcal{M}$
14: $\quad\quad$ **end if**
15: $\quad$ **end while**
16: $\quad$ **return** $\mathcal{M}$
17: **end procedure**

---

6) Harmonic Mean (F value): Finally, the harmonic mean of R and P, we call it F value,

$$\text{F} = \frac{2 \cdot \text{R} \cdot \text{P}}{\text{R} + \text{P}}$$

is calculated.

B. Multi Page Evaluation

Since the dataset is very heterogeneous, each page is evaluated on its own. The average is calculated for this page-wise results. This prevents an overbalance of pages with dozens of baselines (like pages containing a table) and yields results representing the robustness of the evaluated algorithms over various scenarios.

C. Examples

Results for different subsets of the GT and HY baselines of Fig. 4a are shown in Tab. II and explained in the following. The small difference between Ex. 1 and Ex. 2 is due to the fact, that in both cases h_1 is aligned to g_2 for the P calculation. Hence, there is no effect on P if g_1 is removed. R is nearly the same, because g_1 and g_2 are both completely covered by h_1. By removing g_2 instead of g_1 (Ex. 3), h_1 is now aligned to g_1 yielding a lower P value, because g_2 covers much more of h_1 than g_1. In Ex. 4 one gets a high P value, because the remaining HY baselines are very well covered by the GT baselines. By adding h_4 (Ex 5) we of course increase R, but decrease P. This is due to the fact that h_3 is aligned to g_4 (as in Ex. 4) and h_4 is not aligned at all and gets a P value of 0.

TABLE II
EXAMPLE VALUES FOR R, P AND F_1 FOR DIFFERENT SUBSETS OF THE GT AND HY BASELINES SHOWN IF FIG. 4A, FOR ALL EVALUATIONS THE TOLERANCE PARAMETER WAS FIXED TO 20.

Ex.	$\mathcal{G}$	$\mathcal{H}$	R	P	F_1
1	$\{g_1, g_2, g_3, g_4\}$	$\{h_1, h_2, h_3, h_4\}$	0.91	0.61	0.73
2	$\{g_2, g_3, g_4\}$	$\{h_1, h_2, h_3, h_4\}$	0.9	0.61	0.73
3	$\{g_1, g_3, g_4\}$	$\{h_1, h_2, h_3, h_4\}$	0.89	0.51	0.65
4	$\{g_1, g_2, g_3, g_4\}$	$\{h_2, h_3\}$	0.35	0.88	0.5
5	$\{g_1, g_2, g_3, g_4\}$	$\{h_2, h_3, h_4\}$	0.43	0.6	0.5

IV. CONCLUSION

We have introduced a new dataset consisting of 2036 pages of archival documents with annotated baselines. A wide span of different times as well as locations is covered. The dataset contains documents with various degradations and complex layouts. Along with the dataset a goal-oriented evaluation scheme based on baseline descriptions is introduced. In our opinion this work provides new challenges as well as a solid basis for competitive evaluations for the document layout community.

ACKNOWLEDGMENT

REFERENCES

[1] B. Gatos, N. Stamatopoulos, and G. Louloudis, "ICDAR2009 handwriting segmentation contest," *International Journal on Document Analysis and Recognition*, vol. 14, no. 1, pp. 25–33, 2011.

[2] A. Antonacopoulos, S. Pletschacher, D. Bridson, and C. Papadopoulos, "ICDAR2009 page segmentation competition," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2009, pp. 1370–1374.

[3] B. Gatos, N. Stamatopoulos, and G. Louloudis, "ICFHR 2010 handwriting segmentation contest," in *Proceedings - 12th International Conference on Frontiers in Handwriting Recognition, ICFHR 2010*, 2010, pp. 737–742.

[4] A. Antonacopoulos, C. Clausner, C. Papadopoulos, and S. Pletschacher, "Historical document layout analysis competition," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2011, pp. 1516–1520.

[5] N. Stamatopoulos, B. Gatos, G. Louloudis, U. Pal, and A. Alaei, "ICDAR 2013 handwriting segmentation contest," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2013, pp. 1402–1406.

[6] M. Murdock, S. Reid, B. Hamilton, and J. Reese, "ICDAR 2015 competition on text line detection in historical documents," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-November. IEEE, aug 2015, pp. 1171–1175.

[7] A. Antonacopoulos, C. Clausner, C. Papadopoulos, and S. Pletschacher, "ICDAR 2013 competition on historical book recognition (HBR 2013)," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2013, pp. 1459–1463.

[8] ——, "ICDAR2015 competition on recognition of documents with complex layouts - RDCL2015," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-Novem, 2015, pp. 1151–1155.

[9] J. Puigcerver, A. H. Toselli, and E. Vidal, "ICDAR2015 Competition on Keyword Spotting for Handwritten Documents," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-November, 2015, pp. 1176–1180.

[10] A. Nicolaou and B. Gatos, "Handwritten text line segmentation by shredding text into its lines," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2009, pp. 626–630.

[11] Z. Shi, S. Setlur, and V. Govindaraju, "A Steerable Directional Local Profile Technique for Extraction of Handwritten Arabic Text Lines Department of Computer Science and Engineering State University of New York at Buffalo," *2009 10th International*, no. i, pp. 0–4, 2009.

[12] A. Garz, A. Fischer, H. Bunke, and R. Ingold, "A binarization-free clustering approach to segment curved text lines in historical manuscripts," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pp. 1290–1294, 2013.

[13] M. Diem, F. Kleber, and R. Sablatnig, "Text line detection for heterogeneous documents," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pp. 743–747, 2013.

[14] R. Saabni, A. Asi, and J. El-Sana, "Text line extraction for historical document images," *Pattern Recognition Letters*, vol. 35, no. 1, pp. 23–33, 2014.

[15] N. Arvanitopoulos and S. Süsstrunk, "Seam Carving for Text Line Extraction on Color and Grayscale Historical Manuscripts," *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, no. Ic, pp. 726 – 731, 2014.

[16] K. Chen, M. Seuret, M. Liwicki, J. Hennebert, and R. Ingold, "Page segmentation of historical document images with convolutional autoencoders," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-November, 2015, pp. 1011–1015.

[17] Y. Boulid, A. Souhar, and M. Elkettani, "Detection of Text Lines of Handwritten Arabic Manuscripts using Markov Decision Processes," *International Journal of Interactive Multimedia and . . .*, pp. 31–36, 2016.

[18] A. Fischer, V. Frinken, A. Fornés, and H. Bunke, "Transcription Alignment of Latin Manuscripts Using Hidden Markov Models," *The Workshop on Historical Document Imaging and Processing (HIP)*, pp. 29–36, 2011.

[19] A. Fischer, A. Keller, V. Frinken, and H. Bunke, "Lexicon-free handwritten word spotting using character HMMs," in *Pattern Recognition Letters*, vol. 33, no. 7, 2012, pp. 934–942.

[20] M. W. A. Kesiman, D. Valy, J.-c. Burie, E. Paulus, I. M. G. Sunarya, K. H. Sok, and J.-m. Ogier, "Southeast Asian palm leaf manuscript images : a review of handwritten text line segmentation methods and new challenges Made Windu Antara Kesiman," *Journal of*, vol. 26, no. 1, 2016.

[21] T. Strauß, T. Grüning, G. Leifert, and R. Labahn, "CITlab ARGUS for Keyword Search in Historical Handwritten Documents: Description of CITlab's System for the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task," ser. CEUR Workshop Proceedings, Évora, Portugal, sep 2016.

[22] A. Graves and J. Schmidhuber, "Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks," *Advances in Neural Information Processing Systems 21, NIPS'21*, pp. 545–552, 2008.

[23] B. Moysset, C. Kermorvant, C. Wolf, and J. Louradour, "Paragraph text segmentation into lines with Recurrent Neural Networks," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-November, pp. 456–460, 2015.

[24] K. Chen, C. L. Liu, M. Seuret, M. Liwicki, J. Hennebert, and R. Ingold, "Page Segmentation for Historical Document Images Based on Superpixel Classification with Unsupervised Feature Learning," in *Proceedings - 12th IAPR International Workshop on Document Analysis Systems, DAS 2016*, 2016, pp. 299–304.

[25] J. Pastor-Pellicer, M. Z. Afzal, M. Liwicki, and M. J. Castro-Bleda, "Complete System for Text Line Extraction Using Convolutional Neural Networks and Watershed Transform," in *Proceedings - 12th IAPR International Workshop on Document Analysis Systems, DAS 2016*, 2016, pp. 30–35.

[26] V. Romero, J. A. Sanchez, V. Bosch, K. Depuydt, and J. De Does, "Influence of text line segmentation in Handwritten Text Recognition," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 2015-November, aug 2015, pp. 536–540.

[27] M. Diem, F. Kleber, S. Fiel, T. Grüning, and B. Gatos, "ScriptNet: ICDAR 2017 Competition on Baseline Detection in Archival Documents (cBAD)," 2017. [Online]. Available: https://doi.org/10.5281/zenodo.257972