cBAD: ICDAR2017 Competition on Baseline Detection

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

Basilis Gatos
Computational Intelligence Laboratory
National Center for Scientific Research Demokritos
bgat@iit.demokritos.gr

Tobias Grüning
Computational Intelligence Technology Lab
University of Rostock
18057 Rostock, Germany
tobias.gruening@uni-rostock.de

***Abstract*—The cBAD competition aims at benchmarking state-of-the-art baseline detection algorithms. It is in line with previous competitions such as the ICDAR 2013 Handwriting Segmentation Contest. A new, challenging, dataset was created to test the behavior of state-of-the-art systems on real world data. Since traditional evaluation schemes are not applicable to the size and modality of this dataset, we present a new one that introduces baselines to measure performance. We received submissions from five different teams for both tracks.**

***Index Terms*—cBAD, baseline detection, text-line detection**

I. Introduction

Baseline detection is considered an open research topic in the document analysis community and is a preprocessing step for e.g. Automated Text Recognition (ATR). The aim of this competition is to evaluate the performance of methods for detecting baselines in archival document images.

A newly created, freely available, real world dataset consisting of 2035 annotated document page images from 9 different archives is the basis of cBAD. Two competition tracks test different characteristics of the methods submitted. Track A [Simple Documents] is published with annotated text regions and tests therefore a method's quality of text line segmentation. The more challenging Track B [Complex Documents] provides only the page area. Hence, baseline detection algorithms need to correctly locate text lines in the presence of marginalia, tables, empty pages, and noise. Figure 1 shows two example document pages of both tracks. Transparent blue areas indicate text regions provided in Track A and page regions in Track B. Blue polylines display the manually annotated baselines.

Previous text line detection competitions such as *ICDAR 2013 Handwriting Segmentation Contest* [1] provide pixel-level groundtruth and use region based error measurements. The *ICDAR 2015 ANDAR Text Lines* [2] competition requires partaking methods to only provide the starting point of text lines. We see cBAD as successor of these competitions with regards to the aim and data modality. However, the dataset and the evaluation protocol are different. Compared to previous competitions, the evaluation set is larger and contains document images with varying layouts, originating from different time periods and locations. Baselines were manually annotated for each text line and a new evaluation scheme is introduced. Evaluating text line detection methods using baselines has on the one hand the advantage that GT production is faster (cheaper) compared to pixel-level annotation and does not require a crucial binarization step. On the other hand, the evaluation is more accurate than comparing text line starting points only. The images are groundtruthed using PAGE XMLs[1] which is commonly used in document analysis.

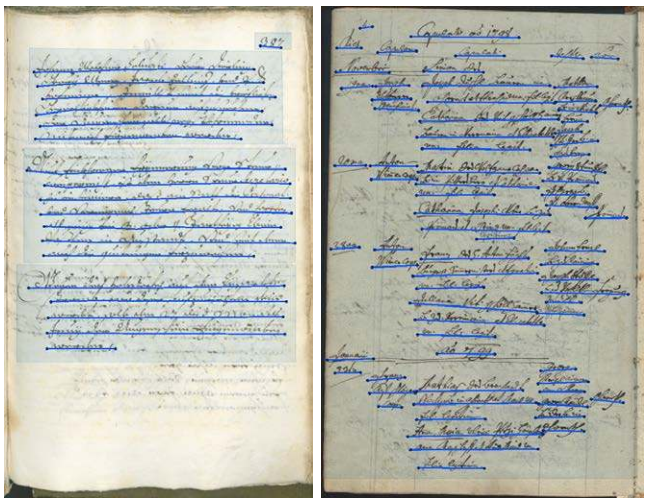

Fig. 1. Two examples of document images of Track A (left) and Track B (right) with annotated baselines and text regions.

Despite of the challenging dataset and the newly introduced evaluation scheme, the competition attracted five teams from

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 674943.

[1] http://www.primaresearch.org/tools

across Europe and the US. We present the dataset, the competition, and the evaluation scheme in the next section. The teams present their respective method in Section III and Section IV presents the results. A short discussion is given in Section V and the paper is concluded in Section VI.

II. The Competition

The ICDAR2017 competition on BAseline Detection (cBAD) dataset consists of 2035 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 characters rest upon and descenders extend below (see Figure 2). 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 between marginalia and the body text (see Figure 3 top).
- they span different columns (see Figure 3).
- they span different document pages (see Figure 3).

If a text line is clearly not part of a table (column) system, a single baseline is annotated (see Figure 3).

Fig. 2. Illustration of a baseline.

B. Database

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[2]. A more detailed description of the document collections can be found in [3].

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[3]. Afterwards, the GT was inspected by two independent operators who removed another 83 images because of wrong baseline annotations resulting in a final dataset size of 2035 images.

[2]https://github.com/TUWien/Benchmarking
[3]https://digi-texx.vn/en/

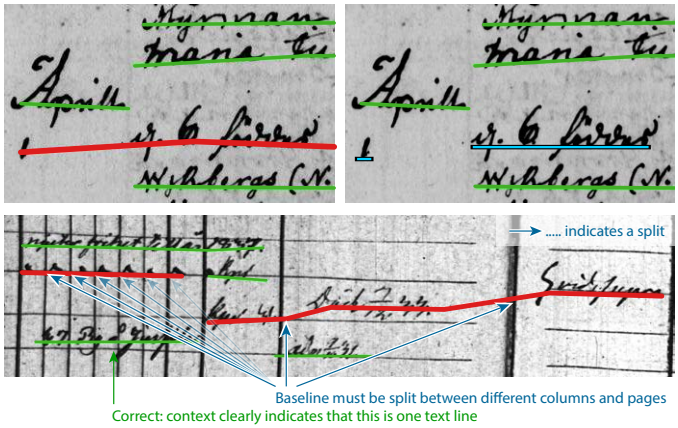

Fig. 3. Example of complex text lines where red (bold) baselines indicate wrongly annotated text lines.

The annotated dataset is split into two tracks: Track A [Simple Documents] and Track B [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. Track B 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 about 30 images are taken from each collection resulting in 216 training images for Track A and 270 images for Track B. The data along with the GT was made publicly available after the end of the competition[4].

The PAGE XML scheme is used for storing text regions and baselines. A minimal sample of a PAGE XML is shown in Listing 1.

C. Evaluation Scheme

Baseline detection is commonly applied prior to ATR which results in these requirements for the evaluation scheme:

- Results should correlate with ATR accuracy (there is not a unique correct baseline, slightly different baselines lead to the same ATR accuracy)
- It should reflect how much of the text was detected (we call this R-value, since it has similar properties as the *recall*)
- It should reflect how reliable the structure of text lines of a document was detected (we call this P-value, since it has similar properties as the *precision*)
- It should not rely on binarization, because there are various algorithms explicitly avoiding binarization [4], [5], [6]
- It must be able to handle skewed and oriented text lines
- It must not rely on the reading order

[4]https://zenodo.org/record/835441

Listing 1. Minimal sample of a PAGE XML containing a text region and a baseline.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<PcGts
  xmlns="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15
  http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15/pagecontent.xsd" >
    <Metadata>
      <Creator>CVL</Creator>
      <Created>2016-10-28T08:46:03Z</Created>
      <LastChange>2017-01-10T10:18:12Z</LastChange>
    </Metadata>
<Page imageFilename="document.tif" imageWidth="2959" imageHeight="4332" >
  <TextRegion id="R0" >
    <Coords points="2401,228 2647,228 2647,399 2401,399"/>
       <TextLine id="L0" >
         <Coords points="2439,306 2574,310 2573,360 2438,356"/>
         <Baseline points="2438,351 2573,355"/>
       </TextLine>
  </TextRegion>
</Page>
</PcGts>
```

We propose a newly developed scheme to evaluate the performance of baseline detection algorithms. It is implemented in Java and publicly available[5] as a standalone command line tool licensed under LGPLv3. A detailed explanation of the evaluation scheme can be found in [3].

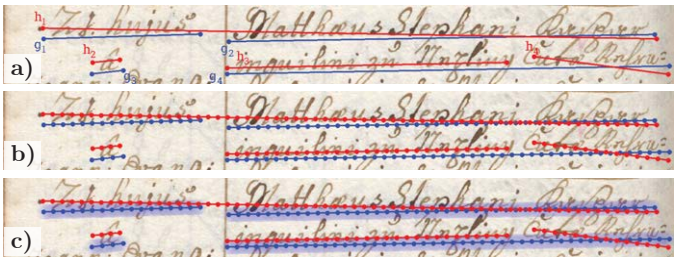

Fig. 4. Baseline sets of four GT baselines (blue) and hypothesis (HY) baselines (red) in a). The normalized polygonal chains in b) (for illustration purposes, every 25th vertex is displayed). GT baselines with tolerance area estimated in c) (here, $t_{\boldsymbol{g}}$ is roughly 20px).

Single Page Evaluation: In the following the calculation of R-value and P-value for a page image 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 page image 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 image (see Fig. 4 a).

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 $\leq \sqrt{2}$) (see Fig. 4 b). 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.

[5]https://github.com/Transkribus/TranskribusBaseLineEvaluationScheme

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. Page (and text line) dependent tolerance values are calculated, because within a collection various resolutions and layout scenarios are present which cannot be covered with a fixed tolerance value. Therefore we estimate the interline distance $d_{\boldsymbol{g}}$ between all GT baselines. The tolerance value is then chosen to be

$$t_{\boldsymbol{g}} = 0.25 \cdot \min(d_{\boldsymbol{g}}, \bar{d}_{\mathcal{G}}).$$

with $\bar{d}_{\mathcal{G}}$ being the average interline distance of a page image. 25% of the estimated interline distance yields a reasonable compromise between accuracy and flexibility.

Coverage Function: A coverage function COV_S counts the number of vertices of a chain $\boldsymbol{p}$ for which there is a vertex of chain $\boldsymbol{q}$ with a distance less than the given tolerance value $t_{\boldsymbol{g}}$. Furthermore a smooth (linear) transition is performed for vertices with a distance between $t_{\boldsymbol{g}}$ and $3t_{\boldsymbol{g}}$.

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

$$\mathrm{R} = \frac{\sum_{\boldsymbol{g} \in \mathcal{G}} \mathrm{COV}_S(\boldsymbol{g}, \mathcal{H}, t_{\boldsymbol{g}})}{|\mathcal{G}|}. \tag{1}$$

The R-value indicates the amount of GT baseline fractions that have corresponding HY baselines within the tolerance area $t_{\boldsymbol{g}}$. 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 $\mathcal{H}$ occurs at most once. The tolerance dependent P-value of $\mathcal{G}$ and $\mathcal{H}$ is then calculated as follows:

$$\mathrm{P} = \frac{\sum_{(\boldsymbol{g}, \boldsymbol{h}) \in \mathcal{M}(\mathcal{G}, \mathcal{H})} \mathrm{COV}(\boldsymbol{h}, \boldsymbol{g}, t_{\boldsymbol{g}})}{|\mathcal{H}|}. \tag{2}$$

The alignment ensures that segmentation errors are penalized. The alignment $\mathcal{M}(\mathcal{G}, \mathcal{H})$ is calculated in a greedy manner which is chosen, because there is no reading order available (no dynamic programming possible) and a greedy solution is in most practical cases the exact solution.

Finally, the harmonic mean of P and R is computed, which we call F-value:

$$\mathrm{F} = \frac{2\mathrm{RP}}{\mathrm{R} + \mathrm{P}} \tag{3}$$

Since the dataset is very heterogeneous, each page image is evaluated on its own. The average is calculated for these 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 approaches over various scenarios.

III. Participants

The competition was carried out using the ScriptNet platform.[6] Teams could download the training set along with GT and the images of the test set. For evaluation, participants uploaded the resulting XMLs (one per image) which were directly evaluated in ScriptNet. Registered teams were able to see the results of their submissions (but results of other teams are hidden). The number of submissions was not limited and the results presented in this paper represent the best submission per team.

Methods of five different teams were submitted for TRACK A and four teams submitted to TRACK B. A short method description provided by the participating teams is given below. They are listed in alphabetical order.

A. BYU

Chris Tensmeyer, Brian Davis, and Curtis Wigington
Dept. of Computer Science, Brigham Young University, Provo, USA
tensmeyer@byu.edu

We formulate the problem of baseline detection as dense pixel classification followed by post processing to correct errors and extract the point representation of the baseline. To classify pixels, we employ a 10-layer Fully Convolutional Network (FCN) that fuses features learned at four image scales. Our objective for Stochastic Gradient Descent (SGD) training is maximizing a continuous relaxation of the traditional F-measure (w.r.t. baseline pixels). We also use truncated distance transforms to compute per-pixel importance weights for both precision and recall, similar to [1]. We trained the FCN using all the provided competition data after downsampling all images by a factor of 4. The ground truth was created by drawing lines with pixel thickness 7 between the baseline points.

To obtain baselines for the simple task from the network output we apply the following post-processing. We first threshold the network output. Then we attempt to detect if the page text is primarily two columns and the x location of the margin by examining the middle portion of the horizontal projection profile of the original document image. If the detected margin cuts through too many connected components of the thresholded network output (6) we discard the detection. We then erode the thresholded network output and use probabilistic Hough lines [2] to detect line segments. We remove line segments with outlier slopes (vertical lines). We cluster the remaining segments such that any lines intersecting the same connected component are in the same cluster. We discard all line segments except the longest from each cluster. We then join the remaining line segments whose lines, if extended, would intersect at a point horizontally between their endpoints. If a margin was detected on the page, we discard any line segments intersecting it. The resulting line segments are drawn onto the thresholded network output with a width of 7 pixels. Connected components are then found on this modified output, and those below a pixel count threshold are removed. We then divide the connected components horizontally into small slices. The centers of mass of these slices, along with the leftmost and rightmost points of the connected component, are the baseline vertices we return.

To obtain baselines for the complex task from the network output we apply the following post-processing. We first threshold the network output and find its connected components. We attempt to detect vertical lines in the original document image by applying a Sobel kernel and finding peaks in the resulting projection profile. We then split any connected components which are divided by the detected horizontal lines such that both halfs are at least 60 pixels long. We then follow the same process as the simple task in removing connected components below a certain pixel count and extracting the baseline vertices.

B. DMRZ

Georg Mackenbrock, Michael Fink, Thomas Layer, Michael Sprinzl
Deutsches Medizinrechenzentrum GmbH & Co KG, Vienna, Austria
mackenb@dmrz.de

Our submission to the cBAD competition utilizes deep convolutional nets as the core means for both, the detection and extraction of baselines from sample images, as well as for the extraction of relevant text regions and the classification of basic document properties (a simple form of layout analysis) in a pre-processing step. For the latter, a convolutional U-Net augmented with auxiliary error layers has been trained on downscaled input images. It returns a mask of regions of interest and, via auxiliary error and output layers, a classification wrt. simple document properties regarding page format and layout. While detected text regions are utilized in TRACK B only, the basic document properties obtained during preprocessing are used in both tracks to parametrize subsequent baseline detection and post-processing (e.g., in computing a scale factor). After simple image pre-processing, candidate baselines are detected by means of a residual U-net (incorporating a slightly modified Dice coefficient). Eventually, track-specific procedural post-processing steps aim at improving the quality of candidate lines (e.g. pruning likely detection errors

[6] https://scriptnet.iit.demokritos.gr/competitions/5/

or joining baseline fragments into a single line) and return a final set of detected lines.

C. IRISA

Aurélie Lemaitre, Jean Camillerapp, and Bertrand Coüasnon
IRISA - University Rennes 2 and Insa Rennes, France
Aurelie.Lemaitre@irisa.fr

The baseline detection submitted to TRACK A (IRISA-A) is based on a blurred image combined with a description of textlines in the context of the document structure for simple textual documents. The baseline detection is a new method using the same blurred image as in the previous method we proposed in [7], but now focused on the lower edges of textlines detected in the blurred image combined with connected components. Hypotheses of textlines produced by this first step are then combined according to a description of textlines defined in EPF, using the DMOS-PI method [8]. This combination build textlines by assembling textlines hypotheses, following rules on their contextual alignments. When available, IRISA-A limits its detection to the global bounding box computed from the XML files. IRISA-A has been applied on Track A (Simple Documents) using the bounding box and directly applied on Track B (Complex Documents) without any information.

D. LITIS

Guillaume Renton, Clément Chatelain, Sébastien Adam, Christopher Kermorvant, and Thierry Paquet
Normandie Univ, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS, 76000 Rouen, France
guillaume.renton@gmail.com

This method is based on fully convolutional networks, a network architecture used in semantic segmentation. Dilated convolutions layers with different rates, from 1 to 4, are used in order to predict for each pixel in a given image whether it belongs to a text line or not. Dilated convolutions are used rather than standard convolutions with deconvolution to limit decreasing resolution with pooling layers. Fully convolutional networks allow to work with variable input sizes, but due to limited gpu memory, the images are reshaped as follows: the largest side of each image is reduced to 608 pixels, and the other side is reduced in order to keep the same ratio between height and width.

Training is made at a core text level. Thus, the system predict pixels regions referring to a line. Baselines are then extracted from those regions using the RDP (Ramer-Douglas-Peucker) algorithm. The system was pre-trained on a dataset made of 8000 handwritten documents, and then trained on the competition training dataset.

E. UPVLC

Moisés Pastor and Lorenzo Quirós
PRHLT research centre. Universitat Politècnica de València
mpastorg@prhlt.upv.es

The baseline detection technique used for our experiments is based on clustering over a set of interest points. Thus, given a set of points pertaining to a handwritten text image, a partition of this set in disjoint clusters, each one defining a baseline. A modified DBScan clustering technique builds the baselines. To discriminate between points belonging to baseline from those from noise, descenders, etc. Extremely Randomized Trees forest is used as classifier.

IV. RESULTS

The evaluation was carried out with the aforementioned evaluation scheme on both tracks. The median F-value of all submissions for TRACK A is 0.89 and 0.76 for TRACK B. This indicates that state-of-the-art baseline detection methods achieve decent results on historical documents given that the layout is simple and text regions are segmented. If complex layout variations are present (e.g. TRACK B), baseline detection is still a challenging task.

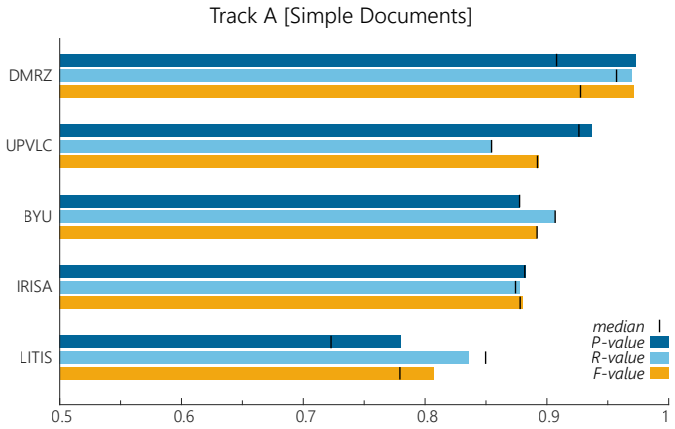

Fig. 5. P-value, R-value, F-value of all submissions of TRACK A. While the bars show the best submission of each team respectively, black lines indicate their median performance.

A. TRACK A *[Simple Documents]*

Figure 5 shows P-value, R-value, and F-value of the best performing submission of each team. All methods are sorted with respect to F-value. Black vertical lines indicate the median performance over all submissions (participants were able to submit their system multiple times with e.g. different parameters). The best performance with an F-value of 0.97 is achieved by the *DMRZ* method (see Table I). The methods submitted by *UPVLC* and *BYU* achieve similar results ($F = 0.89$). The P-value and R-value indicate that the *UPVLC* method splits baselines more precisely but also misses more baselines compared to *BYU*.

B. TRACK B *[Complex Documents]*

As previously mentioned, TRACK B is more challenging which is indicated by the overall performance decrease. Again, the method submitted by *DMRZ* performs best with an F-value of 0.86. In this challenge, the ranking of the other teams changes compared to TRACK A. The F-value of the

Method	P-value	R-value	F-value	Rank
DMRZ	0.973	0.970	0.971	1
UPVLC	0.937	0.855	0.894	2
BYU	0.878	0.907	0.892	3
IRISA	0.883	0.877	0.880	4
LITIS	0.780	0.836	0.807	5

TABLE I
RESULTS ACHIEVED ON TRACK A.

method submitted by *BYU* drops only by 0.1 which is the 2nd best performance achieved in this competition. The *UPVLC* method maintains a high P-value and therefore an accurate segmentation at the cost of missing more text lines than in TRACK A.

Method	P-value	R-value	F-value	Rank
DMRZ	0.854	0.863	0.859	1
BYU	0.773	0.820	0.796	2
IRISA	0.692	0.772	0.730	3
UPVLC	0.833	0.606	0.702	4

TABLE II
RESULTS ACHIEVED ON TRACK B.

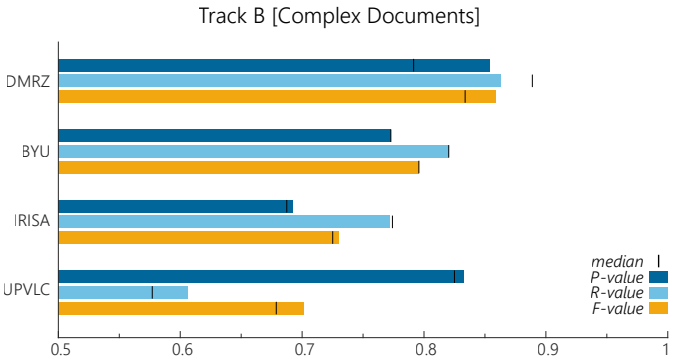

Fig. 6. P-value, R-value, F-value of all submissions of TRACK B. While the bars show the best submission of each team respectively, black lines indicate their median performance.

V. DISCUSSION

The cBAD competition had an open protocol. Hence, participating teams could improve and resubmit their method. After each submission, they received a comprehensive evaluation protocol. This strategy is important to allow for eliminating bugs (e.g. when writing the result files). However, it also allows participating methods to tune their algorithms with respect to the dataset. Figure 7 shows the F-value of each submission grouped by team. For future competitions, we recommend to keep the protocol open but limit the total number of submissions allowed.

VI. CONCLUSION

In this paper, we have evaluated five state-of-the-art baseline detection methods. We introduced two challenging datasets which contain heterogeneous document layouts from different

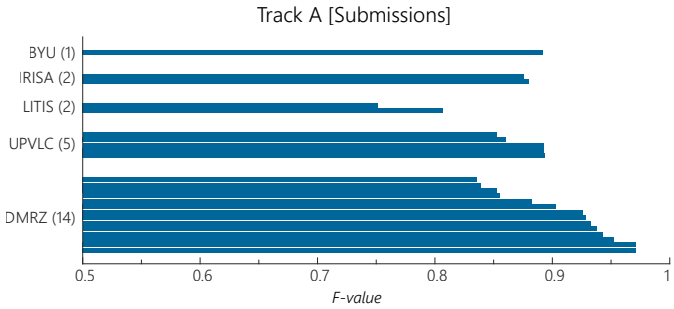

Fig. 7. F-value of TRACK A all individual submissions grouped by team.

sources. The evaluation shows that the method submitted by *DMRZ* achieves the highest performance on both datasets followed by UPVLC in TRACK A and BYU in TRACK B.

We keep the submission system open on ScriptNet[7] which allows for comparing methods developed in the future with those published in this paper. Moreover, both datasets are publicly available which should stimulate future development in the context of baseline detection.

REFERENCES

[1] N. Stamatopoulos, B. Gatos, G. Louloudis, U. Pal, and A. Alaei, "ICDAR 2013 handwriting segmentation contest," in *2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, August 25-28, 2013*. IEEE Computer Society, 2013, pp. 1402–1406. [Online]. Available: https://doi.org/10.1109/ICDAR.2013.283

[2] M. Murdock, S. Reid, B. Hamilton, and J. Reese, "ICDAR 2015 competition on text line detection in historical documents," in *13th International Conference on Document Analysis and Recognition, ICDAR 2015, Nancy, France, August 23-26, 2015*. IEEE Computer Society, 2015, pp. 1171–1175. [Online]. Available: https://doi.org/10.1109/ICDAR.2015.7333945

[3] T. Grüning, R. Labahn, M. Diem, F. Kleber, and S. Fiel, "READ-BAD: A new dataset and evaluation scheme for baseline detection in archival documents," *CoRR*, vol. abs/1705.03311, 2017. [Online]. Available: http://arxiv.org/abs/1705.03311

[4] A. Garz, A. Fischer, and H. Bunke, "A Binarization-Free Clustering Approach to Segment Curved Text Lines in Historical Manuscripts," *2013 12th International*, 2013.

[5] 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.

[6] 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.

[7] A. Lemaitre, J. Camillerapp, and B. Coüasnon, "Handwritten text segmentation using blurred image," in *Document Recognition and Retrieval XXI, San Francisco, California, USA, February 5-6, 2014.*, ser. SPIE Proceedings, B. Coüasnon and E. K. Ringger, Eds., vol. 9021. SPIE, 2014, pp. 90 210D–90 210D–12. [Online]. Available: https://doi.org/10.1117/12.2035735

[8] ——, "Interest of perceptive vision for document structure analysis," in *Human Vision and Electronic Imaging XV, part of the IS&T-SPIE Electronic Imaging Symposium, San Jose, CA, USA, January 18-21, 2010, Proceedings*, ser. SPIE Proceedings, B. E. Rogowitz and T. N. Pappas, Eds., vol. 7527. SPIE, 2010, p. 752714. [Online]. Available: https://doi.org/10.1117/12.838453

[7]https://scriptnet.iit.demokritos.gr/competitions/5/