Formalization and Preliminary Evaluation of a Pipeline for Text Extraction from Infographics

Falk Böschen[1] and Ansgar Scherp[1,2]

[1] Kiel University, Kiel, Germany
[2] ZBW - Leibniz Information Centre for Economics, Kiel, Germany
{fboe,asc}@informatik.uni-kiel.de

Abstract. We propose a pipeline for text extraction from infographics that makes use of a novel combination of data mining and computer vision techniques. The pipeline defines a sequence of steps to identify characters, cluster them into text lines, determine their rotation angle, and apply state-of-the-art OCR to recognize the text. In this paper, we formally define the pipeline and present its current implementation. In addition, we have conducted preliminary evaluations over a data corpus of 121 manually annotated infographics from a broad range of illustration types such as bar charts, pie charts, and line charts, maps, and others. We assess the results of our text extraction pipeline by comparing it with two baselines. Finally, we sketch an outline for future work and possibilities for improving the pipeline.

Keywords: infographics · OCR · multi-oriented text extraction · formalization

1 Introduction

Information graphics (short: *infographics*) are widely used to visualize core information like statistics, survey data or research results of scientific publications in a comprehensible manner. They contain information that is *frequently not present in the surrounding text* [3]. Current (web) retrieval systems do not consider this additional text information encoded in infographics. One reason might be the varying properties of text elements in infographics that makes it difficult to apply automated extraction techniques. First, information graphics contain text elements at various orientations. Second, text in infographics varies in font, size and emphasis and it comes in a wide range of colors on varying background colors.

Therefore, we propose a novel infographic processing pipeline that makes use of an improved combination of methods from data mining and computer vision to find and recognize text in information graphics. We evaluate on 121

 In: R. Bergmann, S. Görg, G. Müller (Eds.): Proceedings of the LWA 2015 Workshops: KDML, FGWM, IR, and FGDB. Trier, Germany, 7.-9. October 2015, published at http://ceur-ws.org

infographics extracted from an open access corpus of scientific publications to demonstrate the effectiveness of our approach. It significantly outperforms two baselines based on the open source OCR engine Tesseract[3].

Subsequently, we discuss the related work. Section 3 presents our pipeline for text extraction and Section 4 specifies the experiment set-up and dataset used. The results regarding our OCR accuracy are presented in Section 5 and discussed in Section 6.

2 Related Work

Research on analyzing infographics is commonly conducted on classifying the information graphics into their diagram type [27] or separating the text from graphical elements [1], [6], [21]. Information graphics show a variety in appearance, which makes such classifications challenging. Thus, many researchers focus on specific types of infographics, e. g., extracting text and graphics from 2D plots using layout information [14]. Other works intend to extract the conveyed message (category) of an infographic [16]. Many research works focus on bar charts, pie charts and line charts when extracting text and graphical symbols [5], reengineer the original data [7], [22], or determine the infographic's core-message [4] to render it in a different modality or make it accessible to visually impaired users.

In any case, one requires clean and accurate OCR results for more complex processing steps, e. g. determining a message. Therefore, they use manually entered text. A different approach [13], [15] to make infographics available to sight impaired users is to translate infographics into Braille, the tactile language, which requires text extraction and layout analysis. This research is similar to our approach but relies on a semi-automatic approach which requires several minutes of human interaction per infographic. Furthermore their approach is challenged by image noise and their supervised character detection algorithm works under the assumption that the text has a unified style, i. e., font, size, and others. Another more specialized approach for mathematical figures [25] describes a pipeline for (mathematical-)text and graphic separation, but only for line graphs and the evaluation corpus is very small and they do not conduct any kind of OCR to verify the results. The assumption to automatically generate high-quality OCR on infographics with today's tools is certainly far-fetched.

3 TX Processing Pipeline

Our Text eXtraction from infographics (short: TX) pipeline consists of five steps plus a final evaluation step as shown in Figure 1. It combines certain ideas from related research [11], [13], [21] to build an automated pipeline which takes an infographic as input and returns all contained text. An initial version of our pipeline was briefly presented in [2]. Here we elaborate in detail on the steps of the pipeline, formalize it, and extend our evaluation. Given the heterogeneous

[3] `https://github.com/tesseract-ocr`, last access: Sep 07, 2015

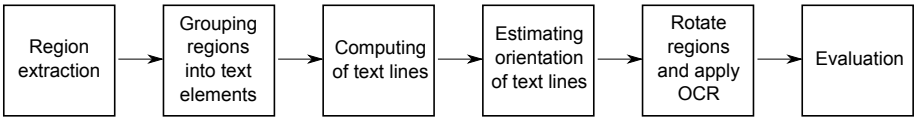

Fig. 1: Novel processing pipeline for text extraction from infographics

research field, a formalization is required to map the related work for a thorough comparison and assessment. In our pipeline, an information graphic I is defined as a set of pixels P with $p = (x, y) \in P \wedge x \in \{1\,..\,width(I)\} \wedge y \in \{1\,..\,height(I)\}$ where the latter two are integer arrays. The color information of each pixel p is defined by a function $\Psi : P \rightarrow S$, where S is a color space. We use this information implicitly during our pipeline and use multiple Ψ functions to map to certain color spaces (e.g. RGB, grey scale,...). A set of text elements T is generated from P by applying the text extraction function Υ:

$$\Upsilon : P, \Psi \rightarrow T \tag{1}$$

Each text element $\tau \in T$ is a sequence of regular expressions ω_i specified as $\tau =< \omega_1, ..., \omega_n >$, separated by blank space characters, and with $\omega =$ [A-Za-z0-9!"§$%&/()=?´°{[]}\‘+-*,.;:|'#@_~<>€éɛ£©®¥¢]*. In the following, we break down the formalization of Υ into five sub-functions υ_j, one function for each step in our pipeline. We define Υ as a composition:

$$\Upsilon := \upsilon_5 \circ \upsilon_4 \circ \upsilon_3 \circ \upsilon_2 \circ \upsilon_1 \tag{2}$$

An overview of the notation used in this paper can be found in Table 1.

Table 1: Symbol notation used in this paper to formalize the TX pipeline

Υ , υ_j	text extraction function Υ and its sub-functions υ_j
P , p	set of pixels P and individual pixel $p \in P$
R , r	set of regions R and individual region $r \in R$
C , c	a clustering C and individual cluster $c \in C$
C' , c'	a set of text lines C' and individual text line $c' \in C'$
Ω , ω	a set of words Ω and individual word $\omega \in \Omega$
A , α	set of text line orientations A and individual orientation $\alpha \in A$
T , τ	set of text elements T and individual text element $\tau \in T$

(1)Region extraction: The first step is to compute a set of disjoint regions R from the infographic's pixel set P using adaptive binarization and Connected Component Labeling [20]. This step is formally defined as:

$$\upsilon_1 : P \rightarrow R, R := \{r | r \subset P \wedge r \neq \emptyset \wedge \forall i, j,\ i \neq j : r_i \cap r_j = \emptyset\} \tag{3}$$

Each region $r \in R$ is a set of pixels forming a connected space, i.e. each region has a single outer boundary, but may contain multiple inner boundaries (holes).

Furthermore, the constraints in equation 3 ensure that all regions are non-empty and disjoint. First, we perform a newly-developed hierarchical, adaptive binarization that splits the infographic into tiles. The novelty of this approach is that it computes individual local thresholds to preserve the contours of all elements. This is based on the assumption that the relevant elements of an infographic are distinguishable through their edges. We start with a subdivision of the original image into four tiles by halving its height and width. For each tile, we apply the popular Sobel operator [24] to determine the edges. We compute the Hausdorff distance [9] over the edges of the current tiles and their parent tile. We further subdivide a tile, by halving its height and width, if a certain empirical value is not reached. A threshold for each tile is computed with Otsu's method [18] and the final threshold per pixel is the average of all thresholds for that pixel. This procedure appeared to be more noise tolerant and outperformed the usual methods, e. g., fixed threshold or histogram, during preliminary tests. The resulting binary image is labeled using the Connected Component Labeling method. This method iterates over a binary image and computes regions based on the pixel neighborhood giving each region a unique label. From the binary image, we compute for each region r the relevant image moments [10] m_{pq} as defined by:

$$m_{pq} = \sum_x \sum_y x^p y^q \Psi \qquad \text{with} \quad p, q = 0, 1, 2, \ldots \tag{4}$$

Please note that p, q hereby denote the p, q^{th} moment and may not be mistaken with the notation used in the remaining paper. For binary images, Ψ takes the values 0 or 1 and therefore only pixels contained in a region are considered for the computation of the moments. Using the first-order moments, we can compute each regions center of mass. Afterwards, we apply simple heuristics to perform an initial filtering. We discard all regions that fulfill the following constraints: (a) Either width or height of the region's bounding box are above average width/height plus 3 times standard deviation (e.g. axes) or (b) bounding box is smaller than 0.001% of the infographic's size (noise) as well as (c) elements occupying more than 80% of their bounding box (e.g. legend symbols). The function υ_1 generates a set of regions R, which can be categorized into "text elements" and "graphic symbols", the two types of elements in an infographic. Thus, in a next step we need to separate good candidates for text elements from other graphical symbols.

(2) Grouping regions to text elements: The second step computes a clustering C from the set of regions R by using DBSCAN [26] on the regions' features:

$$\upsilon_2 : R \to C, \quad C := \{c \subseteq R | c \neq \emptyset \wedge \forall i, j,\ i \neq j : c_i \cap c_j = \emptyset\} \tag{5}$$

Each cluster $c \in C$ is a subset of the regions R and all cluster are disjoint. For each region, the calculated feature vector comprises the x/y-coordinates of the region's center of mass, the width and height of its bounding box, and its mass-to-area ratio. Due to the huge variety of infographics, we apply the density-based hard clustering algorithm DBSCAN to categorize regions into text elements or noise (graphic symbols and others). This step outputs a clustering C where each

cluster is a set of regions representing a candidate text element. We assume that these cluster contain only text while all graphical symbols are classified as noise.

(3) Computing of text lines: In this step, we generate a set of text lines C' on the clustering C by further subdividing each cluster $c \in C$. A text line c' is a set of regions that forms a single line, i.e. the OCR output for these regions is a single line of text. Each clustering c instead may generate multiple lines of text when processed by an OCR engine and therefore may implicitly contain other white space characters. To this end, we apply a second clustering based on a Minimum Spanning Tree (MST) [26] on top of the DBSCAN results, since clusters created by DBSCAN do not necessarily represent text lines. We compute a forest of Minimum Spanning Trees, one MST for each DBSCAN cluster. By splitting up the MST, a set of text lines for each cluster will be built. The rationale is that regions belonging to the same text lines a) tend to be closer together (than other regions) and b) the edges between those regions are of similar orientation. This is defined as:

$$\upsilon_3 : C \rightarrow C', \quad C' := \{c' \subseteq c | c \in C \wedge c' \neq \emptyset \wedge \forall i, j, \ i \neq j : c'_i \cap c'_j = \emptyset\} \tag{6}$$

Each text line $c' \in C'$ contains a subset of the regions of a specific cluster $c \in C$. Again, all text lines are non-empty and disjoint. For each cluster, the MST is built using the regions' center of mass coordinates which are the first two elements of the feature vectors computed in Step 2. We compute a histogram over the angles between the edges in the tree and discard those edges that differ from the main orientation. The orientation outliers are estimated from the angle histogram by finding the maximal occurring orientation and defining an empirical estimated range of ± 60 degrees, where everything outside is an outlier.

(4) Estimating the orientation of text lines: In Step 4, we compute an orientation $\alpha \in A$ for each text line $c' \in C'$ so that we can rotate each line into horizontal orientation for OCR. This can be formalized as:

$$\upsilon_4 : C' \rightarrow C' \times A, \quad A := \mathbb{Z} \cap [-90, 90] \tag{7}$$

Every orientation angle $\alpha \in A$ for a text line c' can have an integer value from -90 to 90 degree. While the MST used in the previous step can well produce potential text lines, it is not well suited for estimating the orientation of text lines as it is constructed on the center of mass coordinates which differ from region to region. Thus, we apply a standard Hough line transformation [12] to estimate the actual text orientation. During the Hough transformation, the coordinates of the center of mass of each element are transformed into a line in Hough space, which is defined by angle and distance to origin, creating a maximal intersection at the lines' orientation. This computation is robust with regard to a small number of outliers that are not part of the main orientation.

(5) Rotate regions and apply OCR: The final step rotates the text lines along an angle of $-\alpha$ in order to apply a standard OCR tool. It is defined as:

$$\upsilon_5 : C' \times A \rightarrow T \tag{8}$$

We cut sub-images from the original graphic using the text lines C' from v_3, rotate them based on their orientation A from v_4 and finally apply OCR.

Step 6, the evaluation of the results, is described in detail below.

4 Evaluation Setup

We assess the results of our pipeline TX by comparing it with two baselines based on Tesseract, a state-of-the-art OCR engine. In our evaluation, we compute the performance over 1-,2- and 3-grams as well as words. During the evaluation, we match the results of TX and the baselines with some gold standard. Both, the position of the text elements as well as their orientation are considered in this process. We use different evaluation metrics as described in Section 4.4.

4.1 Dataset and Gold Standard

Our initial corpus for evaluating our pipeline consists of 121 infographics, which are manually labeled to create our gold standard. Those 121 infographics were randomly retrieved from an open access corpus of 288,000 economics publications. 200,000 candidates for infographics were extracted from these publications. All selected candidates have a width and height between 500 and 2000 pixel, since images below 500 most likely do not contain text of sufficient size and images above 2000 pixel appear to be full page scans in many cases. From the candidate set, we randomly picked images - one at a time - and presented them to a human viewer to confirm that it is an infographic. We developed a labeling tool to manually define text elements in infographics for the generation of our gold standard. For each text element we recorded its position, dimension, rotation and its alpha-numeric content. Please note that we considered using existing datasets like the 880 infographics from the University of Delaware[4], but they were incomplete or of poor quality.

4.2 Baselines

Today's tools are incapable of extracting text from arbitrary infographics. Even approaches from recent research works, as presented in Section 2, are too restrictive to be applicable on information graphics in general. This holds also for specialized research like rotation-invariant OCR [17], [19]. Since no specialized tools exist that could be used as a baseline, we rely on Tesseract, the state-of-the-art OCR engine, as our initial baseline (BL-1). It is reasonable to use this baseline, since Tesseract supports a rotation margin of $\pm 15°$ [23] and is capable of detecting text rotated at $\pm 90°$ due to its integrated layout analysis. Since infographics often contain text at specific orientations ($0°$,$\pm 45°$,$\pm 90°$), we also apply a second baseline. This second baseline (BL-2) consists of multiple runs of Tesseract with the rotated infographic at the above specified angles. We combine the five results from the different orientations by merging the results between those sets and in case of overlaps we take the element with greatest width.

[4] `http://ir.cis.udel.edu/~moraes/udgraphs/`, last access: Sep 07, 2015

4.3 Mapping to Gold Standard

The most accurate approach to compare OCR results with the gold standard would be to evaluate the results on the level of individual characters. Our pipeline, the baselines and the gold standard generate their output on varying levels. Only our pipeline supports the output of individual character regions. Tesseract supports only words, as specified in the hOCR standard[5], on the lowest level. Thus, we transform the gold standard and pipeline output to word level under the assumption of equality in line height and character width. Each text element is defined by its position, i.e. x/y coordinates of the upper left corner of the bounding box , its dimensions determined by width and height of the bounding box and its orientation in terms of a rotation angle around its center. We subdivide each text element τ into words by splitting at blank spaces and carriage returns. The new position and dimensions for each word $\omega \in \Omega$ are computed while retaining the text element's orientation. This is defined by:

$$\Phi: \quad T \times C' \times A \rightarrow \Omega \times C'' \times A \tag{9}$$

$$\Omega := \{\omega \in \tau | \tau \in T\} \tag{10}$$

$$C'' := \{c'' \subseteq c' | c' \in C' \wedge c'' \neq \emptyset \wedge \forall i, j,\ i \neq j : c''_i \cap c''_j = \emptyset\} \tag{11}$$

The bounding boxes of the individual words are matched between TX and gold standard as well as baselines and gold standard for evaluation. For each word $\omega \in \Omega$ we compute the contained n-grams for further evaluation.

4.4 Evaluation Metrics

As previously mentioned, we are evaluating our pipeline over n-grams and words. Since infographics often contain sparse and short text as well as short numbers, we only use 1-,2-, and 3-grams. We use standard metrics precision (PR), recall (RE), and F_1-measure (F_1) for our n-grams evaluation as defined by:

$$PR = \frac{|Extr \cap Rel|}{|Extr|},\ RE = \frac{|Extr \cap Rel|}{|Rel|},\ F_1 = \frac{2 \cdot PR \cdot RE}{PR + RE} \tag{12}$$

Here, $Extr$ refers to the n-grams as they are computed from text elements that are extracted from an infographic by TX and the baseline, respectively. Rel refers to the relevant n-grams from the gold standard. For comparing individual words (i. e. sequences of alpha-numeric characters separated by blank or carriage return), we use standard Levenshtein distance. The same n-gram can appear multiple times in both the extractions result from TX, the baselines, as well as the gold standard. Thus, we have to deal with multisets when computing our evaluation metrics. In order to accommodate this, we have to slightly modify the standard definitions of PR and RE, respectively. To properly account for the number of times an n-gram can appear in $Extr$ or Rel, we define the counter

[5] The hOCR Embedded OCR Workflow and Output Format: `http://tinyurl.com/hOCRFormat`, last access: Sep 07, 2015

function $\mathbf{C}_M(x) := |\{x|x \in M\}|$ (as an extension of a set indicator function) over a multiset M. For an intersection of multisets M and N, the counter function is formally defined by:

$$\mathbf{C}_{M\cap N}(x) := \min\{\mathbf{C}_M(x), \mathbf{C}_N(x)\} \tag{13}$$

Based on $\mathbf{C}_{M\cap N}(x)$, we define PR and RE for multisets:

$$PR = \frac{\sum_{x\in Extr\cup Rel} \mathbf{C}_{Extr\cap Rel}(x)}{\sum_{x\in Extr} \mathbf{C}_{Extr}(x)} \tag{14}$$

$$RE = \frac{\sum_{x\in Extr\cup Rel} \mathbf{C}_{Extr\cap Rel}(x)}{\sum_{x\in Rel} \mathbf{C}_{Rel}(x)} \tag{15}$$

Specific cases may happen when either one of the sets $Extr$ or Rel is empty. One case is that our pipeline TX or the baselines do not extract text where they should, i. e., $Extr = \emptyset$ and $Rel \neq \emptyset$. When such a false negative happens, we define $PR := 0$ and $RE := 0$ following Groot et al. [8]. For the second situation, when the approaches we compare find something where they shouldn't (false positives), i. e., $Extr \neq \emptyset$ and $Rel = \emptyset$, we define $PR := 0$ and $RE := 1$.

5 Results

This section presents the results of our initial evaluation to assess the quality of the OCR results using our pipeline. We start with a descriptive statistics of the gold standard and the extraction results over the infographics. Subsequently, we present the evaluation results in terms of precision, recall and F_1-measure for infographic and word-level evaluation of TX and the two baselines as well as the Levenshtein distances computed for the extracted text and the gold standard.

Data Characteristics: Table 2 presents the average numbers and standard deviation (in brackets) with regard to n-grams, words and word length for our extraction pipeline (TX), both baselines (BL-1/-2), and gold standard (GS). Table 2 clearly shows that our novel pipeline detects at least 1.5 as many n-grams and words as BL-1 and still some more than BL-2. Compared with the gold standard, TX extracts more n-grams and words. In addition TX and the baselines extract words shorter than the gold standard. Overall, we observe high standard deviations in the gold standard and the extraction results.

Evaluation results on word-level n-grams: The average precision (PR), recall (RE) and F_1-measures for n-grams in Table 3 (standard deviation in brackets) show a relative improvement (Diff.) of TX over BL-1 of about 30% on average. The differences are computed by setting the pipeline results into relation with the baselines. We verified the improvement using significance tests, i.e., if the two distributions obtained from TX and BL-1/2 significantly differ. We checked whether the data follows a normal distribution and has equal variances. Subsequently, we have applied Student's t-tests or the non-parametric Wilcoxon

Table 2: Average number of n-grams and words of the 121 infographics and average word length for GS/TX/BL-1/BL-2

	1-grams	2-grams	3-grams	Words	Length
GS	150.65 (122.28)	115.93 (103.09)	84.95 (85.61)	35.46 (22.24)	4.22 (1.48)
TX	177.21 (128.21)	127.34 (100.51)	89.34 (79.35)	50.07 (31.95)	3.63 (2.69)
BL-1	106.30 (87.71)	80.17 (69.12)	60.79 (54.54)	25.21 (22.12)	4.15 (2.25)
BL-2	135.08 (125.56)	100.20 (98.20)	75.08 (78.10)	35.25 (33.94)	4.08 (1.95)

signed rank test. For all statistical tests, we apply a standard significance level of $\alpha = 5\%$. All TX/BL-1 comparison results are significant with $p < .01$ except for the recall over trigrams which has $p < 0.046$. The test statistics for t-tests are between -7.5 and -3.1 and for the Wilcoxon tests between 1808 and 2619. The second part of Table 3 reports the comparison between TX and BL-2. The results are similar to the previous comparison, but for recall over unigrams and F_1-measure over trigrams the improvement is smaller. Here, all differences are significant with a p-value of $p < .01$ except for the recall and F_1-measure over trigrams with $p < 0.049$ and $p < 0.027$, respectively. The test statistics for t-tests are between -6.8 and -3.1 and between 1652 and 2626 for non-parametric tests. Finally, we observe a smaller performance increase when comparing the results from 1-grams to 3-grams as well as overall high standard deviations.

Table 3: Average PR, RE, F_1 measures for TX and BL-1/BL-2

		word level			infographic level		
	n-gram	PR	RE	F_1	PR	RE	F_1
TX	1	.50 (0.41)	.68 (0.36)	.47 (0.39)	.67 (0.23)	.79 (0.20)	.71 (0.21)
	2	.58 (0.39)	.54 (0.38)	.54 (0.34)	.60 (0.27)	.67 (0.25)	.62 (0.25)
	3	.52 (0.39)	.48 (0.37)	.49 (0.37)	.57 (0.29)	.60 (0.29)	.57 (0.28)
BL-1	1	.37 (0.36)	.48 (0.36)	.36 (0.35)	.67 (0.29)	.54 (0.31)	.58 (0.30)
	2	.42 (0.33)	.42 (0.34)	.42 (0.33)	.60 (0.33)	.50 (0.33)	.53 (0.32)
	3	.42 (0.31)	.42 (0.31)	.36 (0.33)	.55 (0.35)	.48 (0.34)	.49 (0.34)
Diff.	1	35.14%	41.67%	30.06%	0.00%	46.30%	22.41%
	2	38.10%	28.57%	28.57%	0.00%	34.00%	16.98%
	3	23.81%	14.29%	36.11%	3.64%	25.00%	16.33%
BL-2	1	.37 (0.37)	.51 (0.38)	.36 (0.36)	.65 (0.25)	.59 (0.29)	.60 (0.26)
	2	.42 (0.34)	.42 (0.35)	.42 (0.34)	.57 (0.31)	.52 (0.31)	.53 (0.30)
	3	.42 (0.32)	.42 (0.32)	.42 (0.32)	.51 (0.33)	.50 (0.34)	.49 (0.32)
Diff.	1	35.14%	33.33%	30.06%	3.08%	33.90%	18.33%
	2	38.10%	28.57%	28.57%	5.26%	28.85%	16.98%
	3	23.81%	14.29%	16.67%	11.76%	20.00%	16.33%

Evaluation results on infographic level n-grams: We conducted another evaluation on infographic level where we did not consider the location mapping

constraint between words and compared the n-grams for the whole infographic. The results are shown in Table 3 for both baselines BL-1 and BL-2. While having on average higher values for all metrics in both comparisons, the relative improvement for precision, recall, and F_1-measure compared with the word level evaluation decreases in most cases. The significance of the results is only given for recall and F_1-measure, but not for precision. For recall and F_1-measure we have $p < .04$ and the test statistics are between -9.2 and -2.4 for t-tests.

Evaluation on words (Levenshtein): For TX the Levenshtein distance is on average 2.23 (SD=1.29). Hence, for an exact match one has to alter about two characters. The average Levenshtein distance for BL-1 is 2.53 (SD=1.59) and we verified that they differ significantly ($t(120) = 2.10, p < .04$). The difference in Levenshtein from BL-2 to TX with an average distance of 2.54 (SD=1.51) is significant as well ($V(120) = 4713, p < .01$).

Special case evaluations: The number of special cases for TX are on average 12.94 (SD=17.88) false negatives and 49.87 (SD=31.52) false positives. For BL-1 we can instead report 17.01 (SD=17.40) false negatives and 5.67 (SD=9.42) false positives on average. BL-2 generates on average 9.03(SD=15.61) false negatives and 17.01(SD=17.40) false positives. Comparing TX pipeline with BL-1 shows that TX produces significantly less false negatives ($V(120) = 4503.5, p < .01$), but simultaneously generates significantly more false positives ($t(120) = -16.6, p < .001$). The second baseline is on average better than TX with regard to false negatives and false positives.

6 Discussion

Our novel pipeline shows promising results for the extraction of multi-oriented text from information graphics. The difference between word and infographic level evaluation can be explained by the constraints induced by the matching procedure on word-level. The main reason for the performance improvement is the increased recall, which is a result of finding text at non-horizontal angles. We define all elements as non-horizontal which have an orientation outside of Tesseract's tolerance range of ± 15 degree. About 20% of the words in an infographic are on average at non-horizontal orientation, as specified by the gold standard. Our pipeline output consists to 37% of non-horizontal words while extracting 41% more words on average than actually present in the gold standard. On the other hand, the first baseline which extracts only about 77% as many words as actually contained, all of horizontal orientation. The second baseline is closest to the gold standard with respect to the number of extracted words and contains on average 31% non-horizontal words. In addition, we have improved precision and therefore an overall performance increase, collected through the F_1-measure, with TX. The standard deviation is in all cases quite high, which can be explained by the variance in the gold standard. Consequently, these are dataset characteristics and not issues of TX or the baselines.

The lower number of 3-grams, which are on average only half as many as 1-grams, is a potential negative influence on the results. As reported in Table 2, there is a high standard deviation of the number of n-grams in the gold standard. Thus, some graphic might not even contain 3-grams. However for most cases, there are on average 85 3-grams per infographic as denoted by the gold standard statistics in Table 2, which is enough for reasonable results.

Furthermore, TX produces less false negatives, i. e., it extracts more text elements from the gold standard than BL-1. But it still makes more mistakes with regard to extracting text elements where there are none in the gold standard. This is reflected in Table 2, where TX extracts on average more text elements than there are actually present in the gold standard. These false positives often consist of special characters such as colons, semicolons, dots, hyphens, and others. Removing them will be a future extension of our work.

7 Conclusion

We have presented our novel pipeline for multi-oriented text extraction from information graphics and proved its concept on a set of 121 infographics. Our text extraction shows a significant increase in F_1-measure over two baselines, which is explained by detecting text elements at non-horizontal angles. In our future work, we plan to add a merge step after the MST clustering to reduce the Levenshtein distance and to perform entity detection over the text extraction results. In addition, we want to apply our pipeline to a larger set of infographics for a more thorough evaluation. We will create the required gold standard using crowd-sourcing in the near future. Finally, we plan to include alternative OCR engines like Ocropus to find the best solution for our needs.

References

[1] P. Agrawal and R. Varma. Text extraction from images. *IJCSET*, 2(4):1083–1087, 2012.

[2] F. Böschen and A. Scherp. Multi-oriented text extraction from information graphics. In *ACM DocEng*, 2015.

[3] S. Carberry, S. Elzer, and S. Demir. Information graphics: an untapped resource for digital libraries. In *SIGIR*, pages 581–588. ACM, 2006.

[4] S. Carberry, S. E. Schwartz, K. F. McCoy, S. Demir, P. Wu, C. Greenbacker, D. Chester, E. Schwartz, D. Oliver, and P. Moraes. Access to Multimodal Articles for Individuals with Sight Impairments. *TiiS*, 2(4):21:1–21:49, 2013.

[5] D. Chester and S. Elzer. Getting Computers to See Information Graphics So User Do Not Have to. In *Foundations of Intelligent Systems*, volume 3488 of *LNCS*, pages 660–668. Springer, 2005.

[6] S. R. Choudhury and C. L. Giles. An architecture for information extraction from figures in digital libraries. In *WWW*, pages 667–672, 2015.

[7] J. Gao, Y. Zhou, and K. E. Barner. VIEW: Visual information extraction widget for improving chart images accessibility. In *ICIP*, pages 2865–2868. IEEE, 2012.

[8] P. Groot, F. van Harmelen, and A. ten Teije. Torture tests: A quantitative analysis for the robustness of knowledge-based systems. In *EKAW*, pages 403–418, 2000.
[9] F. Hausdorff. *Grundzüge der Mengenlehre*. AMS Chelsea Publishing Series. Chelsea Publishing Company, 1949.
[10] M. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962.
[11] W. Huang and C. L. Tan. A system for understanding imaged infographics and its applications. In *ACM DocEng*, pages 9–18, 2007.
[12] J. Illingworth and J. Kittler. A survey of the hough transform. *Computer Vision, Graphics, and Image Processing*, 44(1):87–116, 1988.
[13] C. Jayant, M. Renzelmann, D. Wen, S. Krisnandi, R. E. Ladner, and D. Comden. Automated tactile graphics translation: in the field. In *ASSETS*, pages 75–82, 2007.
[14] S. Kataria, W. Browuer, P. Mitra, and C. L. Giles. Automatic extraction of data points and text blocks from 2-dimensional plots in digital documents. In *Advancement of Artificial Intelligence*, pages 1169–1174. AAAI, 2008.
[15] R. E. Ladner, M. Y. Ivory, R. Rao, S. Burgstahler, D. Comden, S. Hahn, M. Renzelmann, S. Krisnandi, M. Ramasamy, B. Slabosky, A. Martin, A. Lacenski, S. Olsen, and D. Groce. Automating tactile graphics translation. In *ASSETS*, pages 150–157, 2005.
[16] Z. Li, M. Stagitis, S. Carberry, and K. F. McCoy. Towards retrieving relevant information graphics. In *SIGIR*, pages 789–792. ACM, 2013.
[17] R. Mariani, M. P. Deseilligny, J. Labiche, and R. Mullot. Algorithms for the hydrographic network names association on geographic maps. In *ICDAR*. IEEE, 1997.
[18] N. Otsu. A threshold selection method from gray-level histograms. *TSMC*, 9(1):62–66, 1979.
[19] P. M. Patil and T. R. Sontakke. Rotation, scale and translation invariant handwritten devanagari numeral character recognition using general fuzzy neural network. *Pattern Recogn.*, 40(7):2110–2117, 2007.
[20] H. Samet and M. Tamminen. Efficient component labeling of images of arbitrary dimension represented by linear bintrees. *IEEE TPAMI*, 10(4):579–586, 1988.
[21] J. Sas and A. Zolnierek. Three-Stage Method of Text Region Extraction from Diagram Raster Images. In *CORES*, pages 527–538, 2013.
[22] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer. ReVision: Automated Classification, Analysis and Redesign of Chart Images. In *UIST*, pages 393–402. ACM, 2011.
[23] R. Smith. A simple and efficient skew detection algorithm via text row accumulation. In *ICDAR*, volume 2, pages 1145–1148, 1995.
[24] I. Sobel. History and definition of the so-called "sobel operator", more appropriately named the sobel-feldman operator. Sobel, I., Feldman, G., "A 3x3 Isotropic Gradient Operator for Image Processing", presented at the Stanford Artificial Intelligence Project (SAIL) in 1968., 2015.
[25] N. Takagi. Mathematical figure recognition for automating production of tactile graphics. In *ICSMC*, pages 4651–4656, 2009.
[26] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., 2005.
[27] F. Wang and M.-Y. Kan. NPIC: Hierarchical synthetic image classification using image search and generic features. In *CIVR*, volume 4071 of *LNCS*, pages 473–482. Springer, 2006.