Character enhancement for historical newspapers printed using hot metal typesetting

Iuliu Konya Stefan Eickeler Christoph Seibert
Fraunhofer IAIS
Sankt Augustin, Germany
{*e-mail: iuliu.konya, stefan.eickeler, christoph.seibert*}*@iais.fraunhofer.de*

***Abstract*—We propose a new method for an effective removal of the printing artifacts occurring in historical newspapers which are caused by problems in the hot metal typesetting, a widely used printing technique in the late 19^{th} and early 20^{th} century. Such artifacts typically appear as thin lines between single characters or glyphs and are in most cases connected to one of the neighboring characters. The quality of the optical character recognition (OCR) is heavily influenced by this type of printing artifacts. The proposed method is based on the detection of (near) vertical segments by means of directional single-connected chains (DSCC). In order to allow the robust processing of complex decorative fonts such as Fraktur, a set of rules is introduced. This allows us to successfully process prints exhibiting artifacts with a stroke width even higher than that of most thin characters stems. We evaluate our approach on a dataset consisting of old newspaper excerpts printed using Fraktur fonts. The recognition results on the enhanced images using two independent OCR engines (ABBYY FineReader and Tesseract) show significant improvements over the originals.**

***Keywords*-character enhancement; OCR; retro-digitization; historical documents; hot metal typesetting**

I. Introduction

The quality of the results of the optical character recognition (OCR) is directly influenced on one side by the quality of the scanning process and by the printing process on the other side. In a digitization workflow the human operator can control the scanning process of the documents and directly take the appropriate measures to deal with scanning errors in the digitization process. By contrast, printing is normally completed a long time before the scanning procedure and is out of the control of the scanning operator performing the retro-digitization. Therefore, the development of algorithms capable of alleviating the problems occurring in the printing process is a highly desirable endeavor.

In this paper we describe such a method for improving the quality of digitized text documents initially produced using letterpress printing. Specifically, we focus on documents printed during the time period spanning from the beginning until the middle of the 20^{th} century. In this period, a very widespread typesetting technique was hot metal typesetting (also known as hot lead typesetting). This technique represented one of the earliest attempts at mechanizing the printing process, which in turn facilitated its use on an industrial scale. The process consisted of injecting molten type metal (with lead as its main constituent) into molds, each having the shape of a single character or a ligature. The obtained sorts were subsequently used to press ink onto paper and produce the print. A common problem with the procedure was the fact that ink tended to infiltrate between the individual sorts upon imbuement, thus producing specific near-vertical artifacts upon pressing them against the paper. As can be seen in figures 1, 4 and 3, such artifacts are quite prominent and have a comparable stroke width as well as the exact same gray level as regular characters in the digitized image. This fact makes it virtually impossible for the artifacts to be removed effectively by using any state-of-the-art global or local thresholding algorithms. A wide selection of thresholding algorithms, alongside with a performance comparison on a pixel-accurate ground truth can be found in the paper of Gatos et al. [1].

As such, we follow a different approach, presented in more detail in section III. Before that however, we give a short overview of related approaches for document and character enhancement. The proposed approach is evaluated on a dataset consisting of old German-language newspaper excerpts via the OCR results obtained from two well-known OCR engines, namely ABBYY FineReader [2] and Google's Tesseract [3]. Conclusions and directions for future work (partly arisen from the practical issues encountered during the evaluation) are present at the end of the paper.

II. Related Work

Much research work has been done in the different areas of degraded document processing. Algorithms for character enhancement are continuously being proposed for coping with all types of printing techniques as well as for handwritten documents. Despite the continued research effort, until now no unified algorithm applicable for all printing techniques exists. Because of the sheer variety of artifacts it is indeed doubtful that such a generic improvement method is actually possible.

One of the most well-researched related areas is the family of bleed-through/show-through removal techniques for double-sided documents. Most similar in intent to the current paper are techniques belonging to the blind family, such as blind source separation [4]. These methods attempt

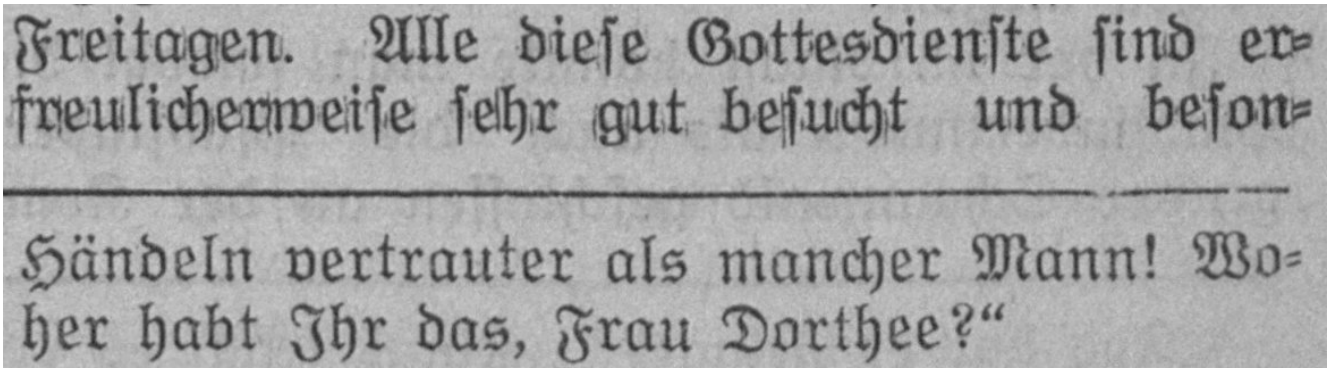


Figure 1. Portion of newspaper image affected by artifacts caused by hot metal typesetting alongside an unaffected area (as appearing in the original print)

to alleviate the bleed-through effects from the document front side without requiring any prior knowledge about its corresponding back side. Unfortunately such an approach cannot be applied in our case, as both the spatial distribution and the gray levels of the printing artifacts are in most cases indistinguishable from those of the regular text.

Another important related research area is that of character enhancement for documents exhibiting various kinds of aging- or printing process-related degradation. In this category fall approaches for character restoration using various energy minimization- and stroke models [5], as well algorithms specialized for the enhancement of low-resolution fax images [6] or typewritten documents [7]. As such methods restrict themselves to strictly improving the visual aspect and connectedness of individual characters they are of limited use in our case.

In [8] the problem of removing printing artifacts from letterpress printed historical newspapers is addressed. The fact that the printing artifacts are thinner than the strokes of the characters is used to remove the artifacts by morphological processing. Successive shrinking and swelling operations with a 5×5 window are performed. The main disadvantage of this method is that artifacts are sometimes worsened by merging the artifact with the character. Another disadvantage is that this method only works if the artifacts are thinner than the strokes of the characters. As can be observed in figure 4, this is not the case in many situations.

III. Printing artifact removal

A. Analysis of printing artifacts and prerequisites

For obtaining an effective text enhancement method it is essential to analyze the nature of the printing artifacts as well as their direct effects beforehand. As mentioned in the introductory section, the targeted artifacts are produced by ink infiltrating between the grouped metal sorts upon imbuement and then leaking onto the paper as the sorts are pressed against the paper sheet during the printing process. Since the traditionally used metal sorts had a rectangular body shape, the ink leaks invariably resulted in near-vertical dashes. More notably because of the fact that hot metal typesetting was designed to automate the printing process, it has found a widespread use starting from the end of the 19^{th} century up until the mid-20^{th} century. As such, most archives containing historical newspapers from the aforementioned period suffer from this kind of artifacts.

Apart from an aesthetically unpleasant appearance, the affected retro-digitized documents suffer from a more accute problem: a low quality OCR result in the text plane. The artifacts, in the form of vertical line segments located between the glyphs are often recognized as spurious “i”s, “I”s, “l”s and sometimes “t”s in the OCR process. Unfortunately, the OCR errors are more severe and difficult to detect as soon as the artifacts intersect the edges of the glyphs. In such cases, due to the similar stroke width of the characters and that of the redundant vertical segments, the characters where the intersection occurs are wrongly recognized by the OCR. Typical examples include the letter “n” recognized as “m”, “c” recognized as “o”, “r” recognized as “n” and “e” recognized as “o”.

In order to correct such errors automatically, the proposed algorithm makes use of and requires information about the textual/font characteristics of the containing text regions. Such information is usually only available in the later stages of the document image analysis (DIA) process. This does not represent a problem however, since the OCR process is typically the very last step in the DIA chain. In the following section we consider that we have as input a skew-free binary image as well as a complete list of text lines found on the given document page. Note that this implicitly assumes that page segmentation has already been performed and a text line detection algorithm (such as the one proposed by Breuel [9]) has been applied on the document regions identified as containing text.

B. Algorithm description

For each input text line, our algorithm applies the processing steps described further in this section. The identification

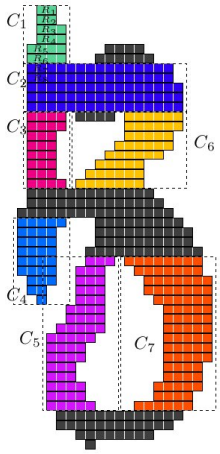


Figure 2. A gothic minuscule "z" and all its contained vertical DSCCs with a height greater than 4 pixels superimposed

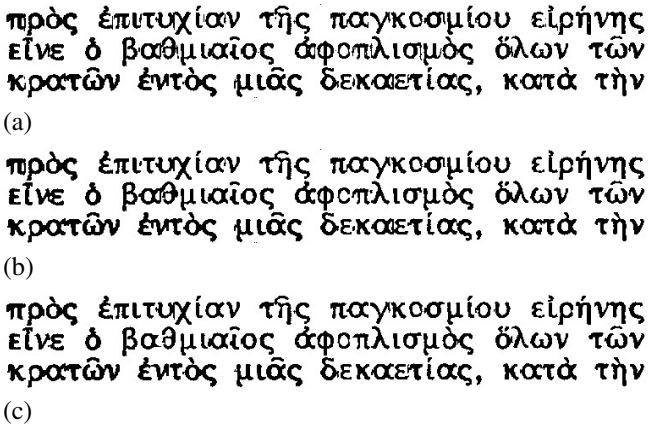


Figure 3. Results of font enhancement: a) original image; b) results of morphological processing, as in [8]; c) results of proposed method. Images a) and b) are taken from [8].

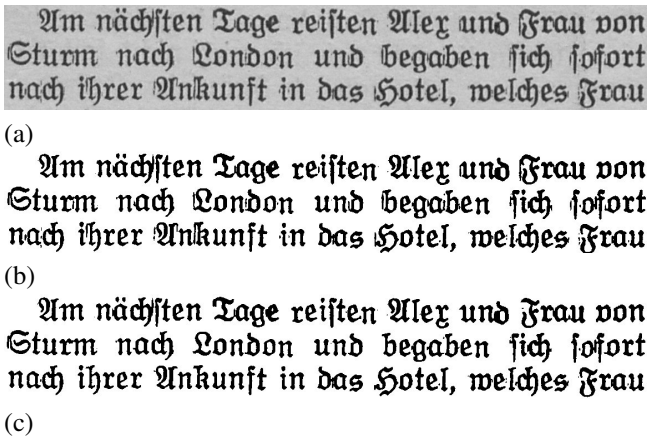
of printing artifacts starts by creating the set of all vertical line segments located within the bounding rectangle of the text line. The vertical line detection employed is based on *horizontal directional single-connected chains* (DSCC). The concept of a DSCC was initially proposed in 2001 by Zhang et al. [10] for the detection of horizontal and vertical lines in printed forms. Here we apply the same algorithm for the detection of vertical line candidates corresponding to possible printing artifacts and located both inside as well as in-between the text characters. A very valuable asset of the DSCC-based detection in comparison to the many other existing line detection algorithms is the pixel-accurate detection of the separators. This allows for an exact removal of the artifacts, should the line segments further on be identified as such.

In general, a vertical DSCC C_V consists of an array $R_1 \dots R_N$ of vertically adjacent horizontal black pixel run-lengths. A horizontal black pixel run-length is defined as a sequence of adjacent black pixels bounded on each side by a white pixel. A DSCC is then formed by a maximal-sized group of neighboring horizontal black pixel run-lengths that have exactly one black run-length neighbor both above and below. In other words, the growth process for a DSCC ends in one direction as soon as either its highest/lowest horizontal black run-length has either zero or more than one black run-length neighbor directly above it/below it. Figure 2 shows the result of the artifact candidate detection process on a single connected component.

For each artifact candidate a *midpoint line fit* is computed, and merging of DSCCs which lie on the same supporting line is performed in the same manner as proposed by Zhang et al (including a minimum height threshold of 4 pixels, which can be adjusted according to the document scanning resolution). For example, in figure 2 only the DSCCs $1-7$ are considered as possible candidates, because all other possible DSCCs are below the height threshold. In addition, we merge all pairs of DSCCs which are located at a larger distance from each other, but having in-between (i.e. on the supporting line) a majority of black pixels. An example of such a situation can be seen in figure 2 for DSCCs C_1 and C_4. Finally, based on the observations in the previous section, we remove from the candidate list all DSCCs deviating from the vertical more than $15°$ and having a height-to-width aspect lower ratio lower than $1:1.5$.

Next, we make use of the *median stroke width* of the characters located on the given text line. The stroke width can be computed exactly and efficiently via a distance transform [11]. As an (less accurate) alternative, one may approximate this value by a simple median of the lengths of the horizontal black run-lengths which were already identified as part of the creation of the DSCCs. In the baseline version of our algorithm (tested in the evaluation section) we keep in the candidate list only those vertical segments with a stroke width strictly lower than the median character stroke width. If it is a-priori known that the text line contains (severe) printing artifacts, this restriction may be relaxed. However, in the case of unaffected text lines, this filtering is essential for reducing the number of false positives. At this point it is worth noting that by depending only relatively to the stroke width we are independent from both the scanning resolution (e.g. in contrast to [8]) and typeface properties such as boldface or italics.

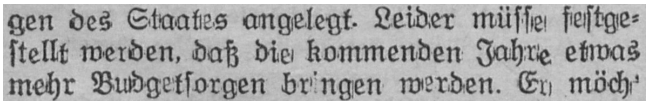
Another essential feature for restricting the list of potential artifact candidates is the *x-height* of the characters forming the text line. We compute the x-height via a 1-dimensional k-Means clustering [13] of the character heights. The number of classes is fixed to 2 (two) and the median height of the characters belonging to the cluster corresponding to the lower height represents our estimate for the x-height. Note that for text lines containing only majuscules or minuscules



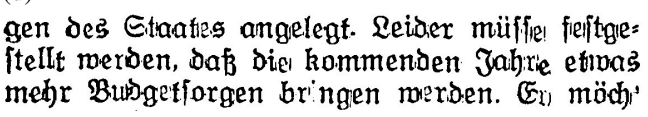
(a)

(b)

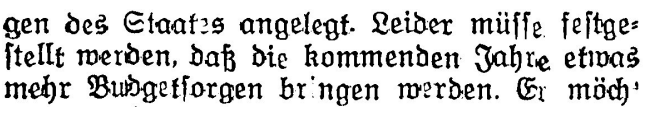
(c)



(d)



(e)



(f)

Figure 4. (a), (d) - portion of original grayscale image; (b), (e) - binarization result using Otsu's method [12]; (c), (f) - result obtained using the proposed method

this approach will obviously not converge to the actual x-height, but in such cases we found it actually desirable to adapt and allow for taller, respectively shorter artifacts. Subsequently, we simulate the deletion of each candidate from its containing connected component. If the height of the resulting component height does not change significantly the candidate is kept, as it likely represents a superfluous protrusion. The other situation when the candidate is kept in the list is when its deletion has cause the connected component to (almost) completely disappear. In the latter case we are very likely dealing with an isolated printing artifact.

At this point the filtering procedure already produces satisfactory results with the notable exception of two cases: the small letter "i" on one side and the set of letters containing counters (e.g. "e", "d", "a", "o", "g", "p"). In both situations the algorithm would identify portions of the respective characters as likely candidates (i.e. either the stem of the "i" or the bowls of the other characters), thus leading to many false positives. The first case can be readily dealt with by searching for a small connected component resembling a dot located right above the candidate. The second case can be identified just as easily, with the sole difference that it additionally involves the extraction of all background connected components from the bitmap of the text line and a straightforward adjacency test.

IV. Evaluation

The evaluation data set consists of 52 single-column text-only excerpts from grayscale newspaper pages printed between 1920 and 1950 and totaling more than 63 000 characters. The newspaper pages originate from the German-language newspaper "Liechtensteiner Volksblatt" and feature a Fraktur script. We have chosen document images printed using this highly decorative script on purpose so as to assess the robustness of the proposed method. As can be seen in figure 3, the proposed technique can readily handle more traditional typefaces, such as Antiqua, as well as non-Romanic scripts. The data set was split about evenly into two groups of images: one containing only text regions clearly affected by printing artifacts and the other containing completely uncorrupted regions. The distinction was done in order to be able to obtain more meaningful evaluation results. This is because a typical newspaper image contains both kinds of regions, irregularly mixed and widely differing in size (number of characters), thus potentially skewing the results greatly in one direction or the other. Also, by having separate evaluation results for affected and unaffected regions one may readily compute a weighted average as an approximation of the expected quality for any image featuring mixed content.

Method	**Affected dataset** **33460 chars** 5034 words 27 images	**Unaffected dataset** **30036 chars** 4730 words 25 images
Tesseract	4793	743
Tess + Enhance	2288	756
Tess + Enhance relative diff.	**52.3%**	**-1.7%**
FineReader	1720	424
FR + Enhance	1074	441
FR + Enhance relative diff.	**37.5%**	**-4%**
Overall relative diff.	**44.9%**	**-2.8%**

Table I
Levenshtein distance [14] from the ground truth without and with the proposed font enhancement method, using Tesseract [3] and ABBYY FineReader [2] as OCR engines

The proposed artifact removal procedure has as primary goal the qualitative improvement of OCR results. Therefore we have chosen as evaluation measure the Levenshtein

distance [14] in conjunction with manually corrected OCR ground truth. In order to ensure that the obtained enhancements are indeed generic and not specific to a particular OCR method, we perform the evaluation using two different, well-known OCR engines: the open source Tesseract v3 [3] and the commercial product ABBYY Finereader 7 [2]. The exact same manually corrected rectangular regions (each corresponding to a single text line) were fed into both OCR engines and the Levenshtein distance was computed on a line-by-line basis. All newspaper images were binarized using Otsu's global thresholding method [12] prior to the application of the enhancement and/or the OCR procedure. Note that the primary purpose of our evaluation is to assess the quality of the proposed enhancement method in the absence of any other external factors. Because of this, document images were specifically chosen so that the scan quality was good enough to allow the application of a global binarization method.

As can be seen from table I our method manages to consistently improve the OCR results of both engines on affected regions. At the same time, the quality loss of the OCR results on uncorrupted text regions is minimal. As such, this is a very encouraging result. In addition, the false positive rate can potentially be reduced even further by observing and appropriately handling the specific situations in which the algorithm still fails. However, we believe that the addition of a (binary) classifier capable of separating corrupted from uncorrupted regions would potentially result in a much more substantial improvement in OCR quality. The prior classification on a region-by-region or line-by-line basis would allow for a more aggressive artifact filtering in heavily affected areas. In both newspaper images from figure 4, one may clearly see that there is still room for improvement especially for thick artifacts intersecting characters.

V. Conclusion

We described a novel method for the automatic removal of letterpress printing artifacts produced as a direct result of problems in the hot metal typesetting. The approach was tested on a dataset consisting of historical newspaper excerpts printed using a Gothic (Fraktur) font and totaling over 63 000 characters. OCR results obtained from two independent state-of-the-art OCR engines (ABBYY FineReader and Google Tesseract) show substantial improvements on document images enhanced via the proposed algorithm. Further research will focus on the automatic detection of affected regions/text lines in order to allow a selective (and consequently much more effective) application of the algorithm.

Acknowledgement

This work was supported by the German Federal Ministry of Economics and Technology (BMWi) funded program *Theseus* in the project *Contentus*.

References

[1] B. Gatos, K. Ntirogiannis, and I. Pratikakis, "ICDAR 2009 document image binarization contest (DIBCO 2009)," in *Proc. Int'l Conf. Document Analysis and Recognition*, 2009, pp. 1375–1382.

[2] ABBYY, "ABBYY FineReader," http://finereader.abbyy.com/, accessed 19 Mar 2011.

[3] Google, "tesseract-ocr," http://code.google.com/p/tesseract-ocr, accessed 19 Mar 2011.

[4] A. Tonazzini, E. Salerno, and L. Bedini, "Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique," *Int'l J. Document Analysis and Recognition*, vol. 10, pp. 17–25, 2007, 10.1007/s10032-006-0015-z. [Online]. Available: http://dx.doi.org/10.1007/s10032-006-0015-z

[5] B. Allier, N. Bali, and H. Emptoz, "Automatic accurate broken character restoration for patrimonial documents," *Document Analysis and Recognition*, vol. 8, no. 4, pp. 246–261, 2006.

[6] J. D. Hobby and T. K. Ho, "Enhancing degraded document images via bitmap clustering and averaging," in *Proc. Int'l Conf. Document Analysis and Recognition*, 1997, pp. 394–400.

[7] A. Antonacopoulos and C. C. Castilla, "Flexible text recovery from degraded typewritten historical documents," in *Proc. IEEE Int'l Conf. on Pattern Recognition*, 2006, pp. 1062–1065.

[8] B. Gatos, S. Mantzaris, S. Perantonis, and A. Tsigris, "Automatic page analysis for the creation of a digital library from newspaper archives," *Digital Libraries*, vol. 3, pp. 77–84, 2000.

[9] T. Breuel, "Two algorithms for geometric layout analysis," in *Proc. Workshop on Document Analysis Systems*, vol. 3697, 2002, pp. 188–199.

[10] Y. Zheng, C. Liu, X. Ding, and S. Pan, "Form frame line detection with directional single-connected chain," in *Proc. Int'l Conf. on Document Analysis and Recognition*. IEEE Computer Society, 2001, pp. 699–703.

[11] C. R. Maurer, R. Qi, and V. Raghavan, "A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions," *Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, 2003.

[12] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[13] J. Hartigan, *Clustering algorithms*. New York: John Wiley and Sons, Inc., 1975.

[14] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.