Line Detection in Binary Document Scans: A Case Study with the International Tracing Service Archives

Benjamin Charles Germain Lee[1,2]

[1] 2017 Digital Humanities Associate Fellow
Levine Institute for Holocaust Education &
Jack, Joseph, & Morton Mandel Center for Advanced Holocaust Studies
United States Holocaust Memorial Museum
Washington, D.C., USA
blee@ushmm.org

[2] Visiting Fellow
Department of History
Harvard University
Cambridge, Massachusetts, USA

***Abstract*—In this short paper, I present my in-progress work on a method of line detection in binary document scans that is capable of differentiating solid and dotted lines. This method entails post-processing candidate lines detected using the progressive probabilistic Hough line transform by filtering out false positives. Solid lines are identified by performing a cut on the average pixel value of the pixels along each candidate line, and dotted lines are identified by performing a cut on the dominant frequency of the Fast Fourier Transform of the same pixel values along each candidate line. I demonstrate the efficacy of this method by running this algorithm on a subset of binary TIF images from the International Tracing Service digitized archives, one of the world's largest collections of Holocaust-related documents. In the case of the International Tracing Service archive, classifying documents based on line structure provides an effective method of extracting information from the documents in an automated fashion, an otherwise intractable endeavor due to low scan quality and the prevalence of handwritten text throughout the archive. My proposed method of identifying line structure represents the first step in this proposed pipeline of classifying International Tracing Service documents by line structure.**

***Keywords*—line detection; dotted lines; computational archival science; International Tracing Service; Holocaust research**

I. Introduction

A ubiquitous challenge in digital archival science is the extraction of metadata from images of documents in digitized collections, a crucial step in improving the accessibility and searchability of collections. Indeed, in the limit of very large document collections, automated methods must be employed in order to extract these metadata in a timely fashion. Handwritten documents and poor resolution document scans further obfuscate this process, as batch extraction of text using optical character recognition is often not possible. Consequently, other methods of extracting metadata must be utilized. One such method is document layout analysis: the segmentation of a document into distinct regions for the purposes of document classification and targeted analysis of specific regions, such as targeted optical character recognition; a survey of literature on document layout analysis can be found in [1]. One approach to document layout analysis is to perform line detection on document images and use the detected lines for segmentation [2]-[6]. Indeed, much work has been done on line detection in document images, including the development of algorithms capable of detecting dashed and dotted lines [7]-[11]. Here, I consider the case of binary images, in which the information loss in the binary pixel encoding manifests not only as the loss of gray colors but also as the degradation of valuable gradient information, making line detection and document layout analysis even more difficult.

II. A Case Study: The International Tracing Service Archives

Established as a tracing service "to help reunite families separated during [World War II] and to trace missing family members," the International Tracing Service (ITS) archive represents one of the world's largest collections of Holocaust-related documents [12]. It contains document subcollections ranging from concentration camp records to correspondences attempting to identify unaccompanied children [13]. Beginning in 2007, an effort to capture over 190 million images representing the collection was undertaken in order to make the archive more accessible both to Holocaust victims and their families and to historical researchers [14]. However, because the archive still serves as a tracing service, the digitized archive itself has been organized and indexed according to a phonetic name system with sparse additional metadata. This lack of metadata restricts the ability to search the digitized archive and has necessitated the search for alternative automated methods of extracting metadata.

Perhaps the most obvious method of extracting metadata from images of documents in the ITS collection is using optical character recognition to read typewritten text. However, an overwhelming majority of the images in the ITS collection are difficult to read using optical character recognition packages for two primary reasons: the images are low resolution binary TIFs, and much of the valuable information on the documents is handwritten, not typewritten.

Fortunately, 47 million cards from the Central Name Index (CNI), produced by the ITS staff to organize the documents in the archive as a finding aid of sorts, have rich line structures by which the cards can be classified [13]. Representative examples of a subset of such card types are depicted at the end of this paper. A card's line structure is particularly useful in this instance because the line structure is associated with other forms of information, such as whether the card refers to a person or to a specific document. Thus, detecting lines on the binary TIF images of cards from the Central Name Index and classifying the cards according to the identified line structure provide a method of extracting a valuable form of metadata from an archive too massive to be searched manually. Furthermore, the classification of the cards using the identified line structure provides a separation of the cards into groups more amenable and less amenable to optical character recognition, as cards with certain types of line structure are less likely to contain handwritten text. Lastly, line detection provides a natural method of segmenting documents into regions for targeted optical character recognition, and in this regard, line detection serves as a method for bootstrapping the extraction of metadata from this document collection. In this paper, I focus solely on the initial progress made with first step of this classification pipeline: the detection of lines on the binary TIF scans of cards from the Central Name Index.

III. Methodology: Robust Line Detection in Binary Scans of Documents

Ubiquitous methods of line detection in images are the Hough line transform and variations thereof.[1] Here, I adopt the progressive probabilistic Hough transform (PPHT) as the first step in line detection [15]; however, the following statements are also true for the Hough line transform. In the case of binary document images, the PPHT suffers from the information loss in restricting each pixel to be either black or white. This information loss is particularly detrimental for the PPHT in the case that the scan is noisy or the scan contrast is poor. Consequently, there is an inherent tradeoff in applying the PPHT in the case of binary document images with poor scan quality: either one must choose parameters for a PPHT that result in many true lines not being detected, or one must choose parameters such that the PPHT results in false-positive candidate lines.[2] While the former is preferable because it does not have such a high proportion of false positives, it is inherently limited by incompleteness. The latter method, on the other hand, is not restricted by such limitations if there is an effective method of eliminating false positives. In this section, I present a two-part post-processing method for filtering false positives that is fairly robust to varying scan conditions. In particular, this filtering process also differentiates solid lines from dotted lines, providing more information about the underlying document structure. Though other methods of identifying differing line types have been explored [7]-[11], this method is noted for its simplicity in only requiring minor postprocessing to the PPHT.

A. *Part I: Identifying Solid Lines*

Given a set of candidate lines returned by the PPHT, the first question becomes how to identify the candidate lines corresponding to true solid lines in the binary scans of documents. I propose a simple filtering method according to the following general observation: pixels corresponding to a true solid line in an image are, on average, more likely to be black than pixels corresponding to a false-positive candidate line, such as a candidate line that runs over text or scan noise. This provides a first-order method of eliminating false-positives: given $\{(x_1, y_1), (x_2, y_2)\}$ coordinates of a line found using the PPHT, these coordinates are mapped to pixel space in order to identify a set of N pixels $\{(p_{x,i}, p_{y,i})\}_{i=1}^{N}$ that fall along the candidate line. The average pixel value is then computed by accessing the pixel value at each such pixel and then averaging over all desired values. In order to account for slight misalignment in the placement of true lines along the transverse axis by the PPHT, I suggest shifting the line locally along the transverse axis, computing the average pixel values for each line along these shifts, and recording the minimum average pixel value of such lines.[3]

This metric decomposes the true solid lines and false-positive solid lines into two fairly distinct groups, the distinctiveness of which is inversely related to scan quality. A simple threshold cut can then be performed on the average pixel values for all candidate lines in order to eliminate false-positive lines with higher average pixel values. For a document with no solid lines, the relative mass of lines with extremely low average pixel values will inevitably be small, making the identification of such a document fairly straightforward.

B. *Identifying Dotted Lines*

Identifying dotted lines in binary scans of documents provides a greater challenge, given that the method described in Section III-A eliminates dotted lines: dotted lines have higher average pixel values because a dotted line by definition consists of periodic stretches of white pixels in between stretches of black pixels. Thus, an alternative method of identifying dotted lines must be considered.

The method that I propose for identifying dotted lines exploits the periodicity of dotted lines. Here, the logic is as follows: a false-positive line may have the same average pixel value as a dotted line, but a false-positive line running over text or scan noise should, in general, exhibit less periodicity. Thus, a measure of the periodicity of the lines should be a reasonable first-order metric for identifying dotted lines. Here, I measure the periodicity of a candidate line by taking the Fast Fourier Transform (FFT) of the pixel values along the candidate line and calculating the dominant frequency of the FFT; a dotted line should have a high corresponding dominant frequency. Analogous to Section III-A, I recommend shifting the candidate line locally along the transverse axis and taking the dominant frequency corresponding to the line with the minimum average pixel value in order to account for small error in line placement with the PPHT. A threshold cut on dominant frequency then identifies dotted lines.

In principle, FFT dominant frequency should also be an effective method of identifying solid lines because solid lines should have a dominant frequency of zero. However, based on empirical evidence using the binary document scans in the ITS document collection in question, average pixel value is a better indicator of solid lines, as described in Section IV. Consequently, this dominant frequency cut in its current form is advocated for the identification of dotted lines only.

IV. Results: Central Name Index Cards from the International Tracing Service Archives

In this section, I present the results of running this line detection algorithm on a small subset of binary TIF scans of

[1] The progressive probabilistic Hough transform [15], the randomized Hough transform, etc.

[2] These false positives include lines that run over text and lines that run over noise from the scan.

[3] I adopt the convention that a black pixel is 0 and a white pixel is 255, and thus, the lowest minimum average pixel value corresponds to the darkest average pixel.

Central Name Index cards from the ITS archive. This section is not intended to be a comprehensive analysis of this method's performance on the Central Name Index cards but rather is intended to serve as a demonstration of the potential for this method, as it is a work-in-progress. For all tests, I used my personal laptop, a 2015 15-inch Macbook Pro with a 2.8 GHz Intel Core i7 processor. I wrote all code in Python and utilized OpenCV for all image processing through the PPHT step. Though this method has by no means been optimized for runtime yet, it requires about 1 second of runtime per image on my laptop; optimizing this algorithm for runtime is left for future work.

To pre-process the images for the PPHT, I first cropped the images by ten pixels along their borders in order to eliminate false-positive lines due to scanner misalignment. I then applied the Canny edge detector with an aperture size of 3 [16]. Next, I processed the images with the PPHT as implemented by OpenCV with the parameters empirically set as follows: minimum line length = 25, maximum line gap = 50, $\rho = 1.25$, $\theta = \pi/100$, and threshold = 50. The results of running the PPHT on a sample image after the Canny edge detector can be seen in Figure 1. In accordance with Section III, these settings were chosen such that the PPHT would identify too many lines rather than too few because the false positives can be filtered out using my proposed method.

Next, I computed the average pixel value for each candidate line using the pixel values in the original image. I computed the dominant frequency for each candidate line using the pixel values in the original image convolved with a Gaussian with a 5×5 pixel kernel because this blurring on average yielded a better decomposition of the dotted lines and the rest of the candidate lines under the metric of FFT dominant frequency, as determined empirically.[4] In computing both the average pixel value and dominant frequency in accordance with Section III, I chose to shift each candidate line along its transverse axis by two pixels in both directions in order to account for slight misalignment with line placement by the PPHT.

In Figure 1, I present a scatter plot of dominant frequency versus average pixel value for candidate lines in a sample Central Name Index card with both solid and dotted lines. As seen in the scatter plot, these metrics do indeed separate the candidate lines into a distinct low average pixel value cluster and a distinct high dominant frequency cluster, at least in the case of a cleanly-scanned image. As one would expect, these clusters start to blend with the false positives more as the scan quality of an image decreases, meaning that the cuts on average pixel value and dominant frequency become less effective.

The procedures for selecting both cut points were tuned using of order hundreds of sample images. For the average pixel value cut point, I used kernel density estimation (KDE) with a bandwidth of one-third of the bandwidth specified by Scott's Rule to fit the average pixel values of all candidate lines for a given image. I then took the cut point to be the left-most minimum with an x-coordinate less than an average pixel value of 70, if one existed (if not, no cut point was taken). In order to handle the case of an image with no solid lines, I computed a 50-bin histogram of average pixel values for the candidate lines in an image and enforced that if the identified cut point was greater than 40, no more than 1/3 of the total lines that fell into bins up to the one containing the cut point could also fall into the five bins just to the left of the bin containing the cut point. If this criterion was not met, then no cut point was taken. This method proved to be robust in eliminating false-positive cut points and handled Central Name Index cards without solid lines effectively, at least when run on several hundred images; an example of this is presented in Figure 2.

One must be more careful in identifying the cut point for dominant frequency than for average pixel value for two reasons: images without dotted lines can still produce large clusters of candidate lines with higher dominant frequencies *relative* to the rest of the identified candidate lines, and the dominant frequencies of dotted lines vary from image to image.[5] Consequently, naively picking the first minimum above a certain threshold does not perform as well in this case. For this reason, I chose the cut point for dominant frequency as follows. First, I produced a histogram of dominant frequencies weighted by line length with 50 bins ranging from a dominant frequency of 0 to a dominant frequency of 0.25 pixels^{-1}.[6] I then identified a cut point by adding each bin to its right neighbor (ignoring the rightmost bin) and identifying the rightmost such bin with a left endpoint greater than 0.08 pixels^{-1} such that the count in the bin was greater than $7,000$. This was tuned empirically using of order hundreds of images, but the general strategy of identifying the last minimum before a peak with high mass in the weighted histogram should in general be a reasonable approach.

In Figures 2, 3, 4, and 5, I present the results of applying these cuts in average pixel value and dominant frequency to 4 example cards. In conjunction with the example in Figure 1, one can see that this method successfully filters out most of the false-positive candidate lines, with some exceptions. In particular, this method fails to remove two primary types of false-positive lines: small candidate solid lines that fall on stretches of text that are entirely black, and candidate dotted lines that run over text and coincidentally happen to have high dominant frequencies due to the periodicity of the white space between the text characters. It is worth noting that many of these false-positive lines can be removed by filtering out lines that are skewed relative to the other lines in the image because the lines on the Central Name Index cards are all either horizontal or vertical relative to the page (but not necessarily aligned to the scan coordinates, as some images in the archive have been scanned at an angle, as seen in Figure 3). This can be accomplished by performing a cut on the modified z score

[4] It is important to note here that a 5×5 pixel kernel is not optimal for identifying dotted lines in all images; for some images, less smoothing performs optimally, and for others, more smoothing performs optimally. This, of course, must be addressed in any final implementation of this line detection algorithm; ordinarily, I would show concrete examples of this phenomenon in this paper, but this paper is intended to be a short overview of this project. One possible solution is to run the line detection algorithm with multiple different kernel sizes on each image and then utilize all of these realizations in the classification algorithm, as different smoothing kernels highlight different physical scales.

[5] This is due to the size of the printed dots, not due to varying pixel size from different scan resolutions.

[6] Weighting the histogram by line length was motivated by the fact that for the ITS CNI cards, one expects dotted lines to be on average longer than false-positive lines, as determined empirically.

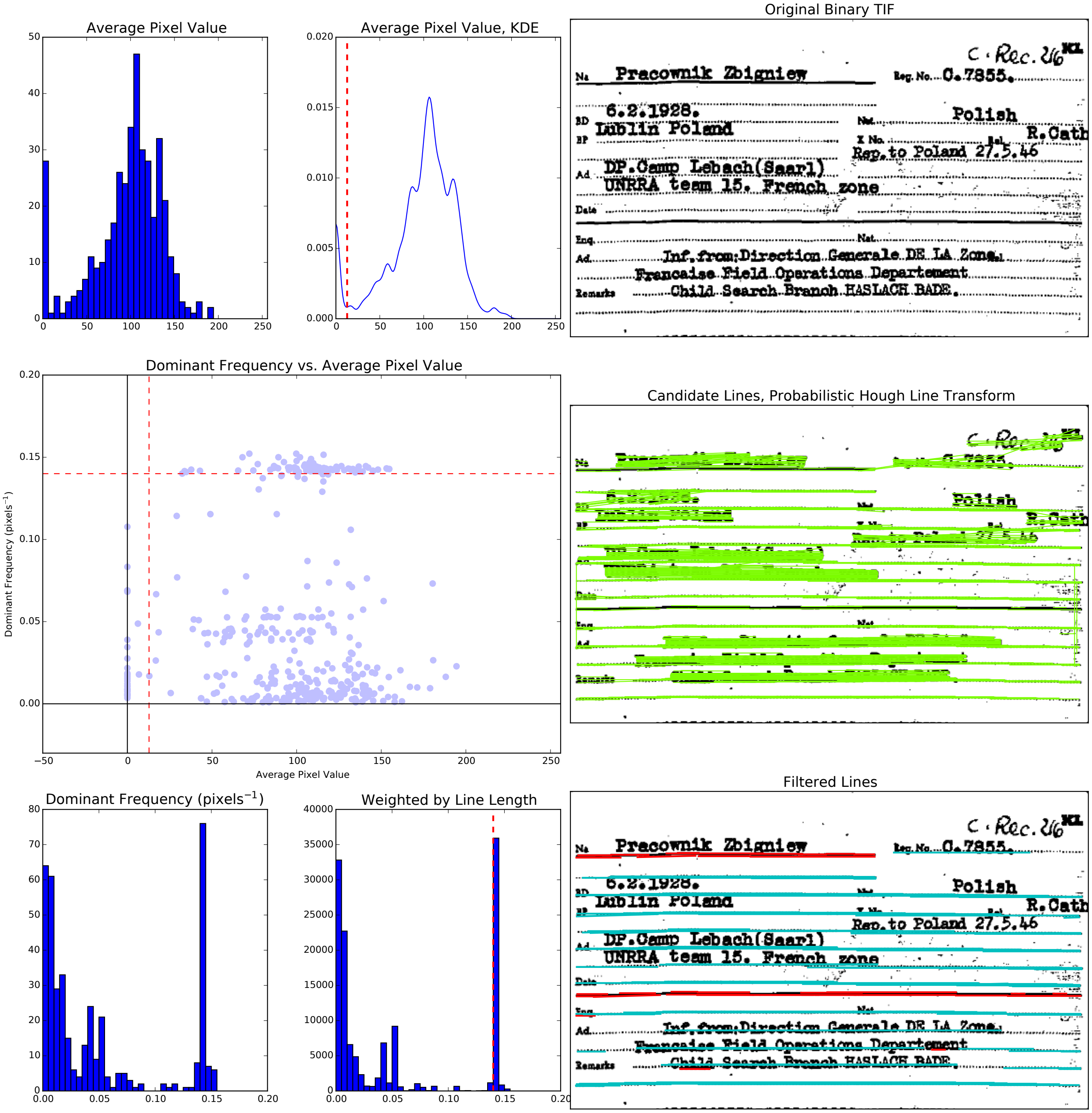


Fig. 1. An eight panel plot showing the line detection algorithm presented in this paper applied to a sample Central Name Index card from the International Tracing Service archives. This card was chosen because it has both solid and dotted lines and therefore serves as a representative example of both cut point selection procedures. In the scatter plot, the cluster of candidate lines with low average pixel values and the cluster of candidate lines with high dominant frequencies can readily be seen. The dotted red lines correspond to the cut points. The two plots on the top left and the two plots on the bottom left are included to elucidate the cut point selection procedure described in Section IV. On the right half of the plot, the top image corresponds to the original binary TIF image of the Central Name Index card, as found within the ITS archive. The middle image shows in green the candidate lines identified by the progressive probabilistic Hough line transform according to the parameters detailed in Section IV. The bottom image shows the results of the candidate line filtering process that constitutes the post-processing step in my line detection algorithm; here, solid lines are plotted in red, and dotted lines are plotted in cyan. A few false-positive solid lines appear over the bottom portions of text, and small regions of dotted lines are filtered out, but overall, this procedure filters solid lines and dotted lines to a high degree of accuracy. *CNI card of Zbigniew Pracownik, 0.1/32699964/ITS Digital Archive, USHMM*

of the skew angles of the candidate lines.[7]

The other failure mode is the case in which a scan is bright, and solid lines are fragmented by the scanning process, as seen in Figure 5. In particular, the cut on average pixel value eliminates any candidate lines that are indeed correct solid lines. These cases are exceptionally difficult to handle, and I am currently working on making this more robust.

V. Future Work

As stated throughout this paper, this line detection algorithm and its application to the International Tracing Service archive are works-in-progress. Once this algorithm is refined further, the next step would be to use this line detection algorithm to classify the Central Name Index card images by document type using a clustering algorithm or machine learning. The exact method by which the detected lines will be incorporated with the raw scan images in the classification algorithm is left for future work. However, it is important to note that this line detection algorithm highlights important features of the document images for classification, and thus, any classification algorithm would almost certainly benefit from this information if incorporated appropriately. Once the documents have been classified, the documents can be segmented using the detected line structure, as well as template matching, enabling targeted OCR of localized regions of text.

In regard to improvements to this line detection algorithm itself, there remain many avenues to explore. One example is exploring alternative methods to cut point selection, such as clustering; indeed, the cut point selection methods presented in Section IV of this paper could be further improved in order to eliminate the small fraction of false positives that are allowed by these cuts, as well as to identify the true lines that are eliminated with these cuts. In addition, understanding better the optimal smoothing kernel for dotted line detection in each image is important future work. Furthermore, other methods of line detection should be explored in order to supplement this method. I have begun modifying the method of histogram of oriented gradients for application in this context but do not yet have any results to share. Lastly, running this line detection algorithm on the tens of millions of cards in the Central Name Index would require optimizing the code for runtime.

Acknowledgment

I would like to thank the United States Holocaust Memorial Museum for supporting my fellowship, which has made this work possible. Furthermore, I would like to thank Michael Haley Goldman, Michael Levy, Elizabeth Anthony, Robert Ehrenreich, Gabriel Pizzorno, and Ryan Kerr for their helpful discussions and support. Lastly, I would like to thank the International Tracing Service.

References

[1] S. Mao, A. Rosenfeld & T. Kanungo. “Document Structure Analysis Algorithms: A Literature Survey,” *Proceedings Volume 5010, Document Recognition and Retrieval X*, Santa Clara, 2003, pp. 197-207.

[2] L. O’Gorman. “The Document Spectrum for Page Layout Analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, Nov. 1993.

[3] T. Kasar, P. Barlas, S. Adam, C. Chatelain & T. Paquet. “Learning to Detect Tables in Scanned Document Images Using Line Information,” *Proc. 12th International Conf. Document Analysis and Recognition*, Washington, D.C., 2013, pp. 1185-1189.

[4] B. Gatos, D. Danatsas, I. Pratikakis & S.J. Perantonis. “Automatic Table Detection in Document Images,” *Proc. International Conf. Pattern Recognition and Image Analysis*, Bath, 2005, pp. 609-618.

[5] L. Likforman-Sulem, A. Hanimyan & C. Faure. “A Hough Based Algorithm for Extracting Text Lines in Handwritten Documents,” *Proc. 3rd International Conf. Document Analysis and Recognition*, Montreal, 1995, pp. 545-549.

[6] G. Louloudis, B. Gatos, & K. Halatsis. “Text Line Detection in Unconstrained Handwritten Documents Using a Block-Based Hough Transform Approach,” *Ninth International Conf. Document Analysis and Recognition*, Parana, 2007, pp. 599-603.

[7] N.S. Rani & T. Vasudev. “An Efficient Technique for Detection and Removal of Lines with Text Stroke Crossings in Document Images,” *Proc. International Conf. Cognition and Recognition*, 2016, pp. 83-97.

[8] Y. Zheng, H. Li, D. Doermann. “A model-based line detection algorithm in documents,” *Proc. of Seventh International Conf. Document Analysis and Recognition*, 2003, pp. 44-48.

[9] B. Kong, S. Chen & R.M. Haralick, “Automatic Line Detection in Document Images Using Recursive Morphological Transforms,” *IS&T/SPIE Symposium on Electronic Imaging: Science and Technology, Proc. Vol 2422, Document Recognition II*, San Jose, 1995, pp. 163-174.

[10] B. Kong, I.T. Phillips, R.M. Haralick, A. Prasad & R. Kasturi. “A Benchmark: Performance Evaluation of Dashed-line Detection Algorithms,” *GREC 1995: Graphics Recognition Methods and Applications*, pp 270-285.

[11] L. Wenyin & D. Dori. “A Generic Integrated Line Detection Algorithm and Its Object - Process Specification,” *Computer Vision and Image Understanding*, vol. 70, no. 3, June, 1998.

[12] The Holocaust Survivors and Victims Resource Center at the United States Holocaust Memorial Museum, “ITS Frequently Asked Questions,” https://www.ushmm.org/remember/the-holocaust-survivors-and-victims-resource-center/international-tracing-service/about-the-international-tracing-service/its-frequently-asked-questions.

[13] Charles-Claude Biederman, *60 Years of History and Benefit of the Personal Documentary Material about the Former Civilian Persecutees of the National Socialist Regime Preserved in Bad Arolsen*.

[14] Senate Resolution 142 (2007): “A resolution observing Yom Hashoah, Holocaust Memorial Day, and calling on the remaining member countries of the International Commission of the International Tracing Service to ratify the May 2006 amendments to the 1955 Bonn Accords immediately to allow open access to the Bad Arolsen archives.”

[15] J. Matas, C. Galambos & J. Kittler, “Robust Detection of Lines Using the Progressive Probabilistic Hough Transform,” *Computer Vision & Image Understanding*, vol. 78, no. 1, pp. 119-137, April 2000.

[16] John Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, no. 6, pp. 679-698, Nov. 1986.

[17] CNI card of Zbigniew Pracownik, 0.1/32699964/ITS Digital Archive, USHMM.

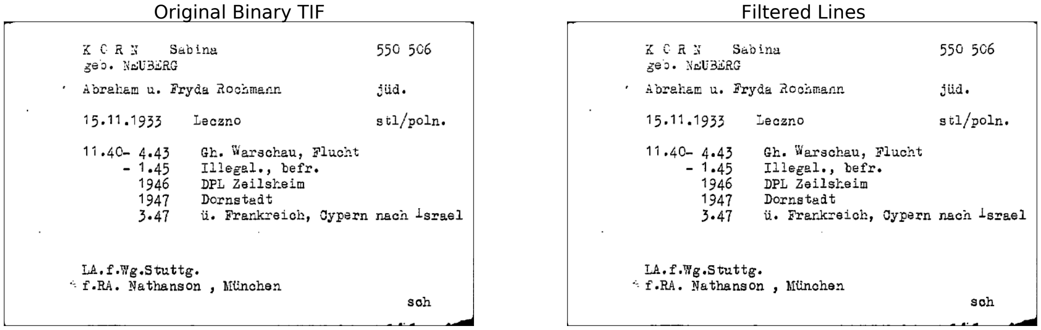
[18] CNI card of Sabina Korn, 0.1/28204178/ITS Digital Archive, USHMM.

[19] CNI card of Miroslav Konecny, 0.1/28003889/ITS Digital Archive, USHMM.

[20] CNI card of Marion Komieczny, 0.1/28004164/ITS Digital Archive, USHMM.

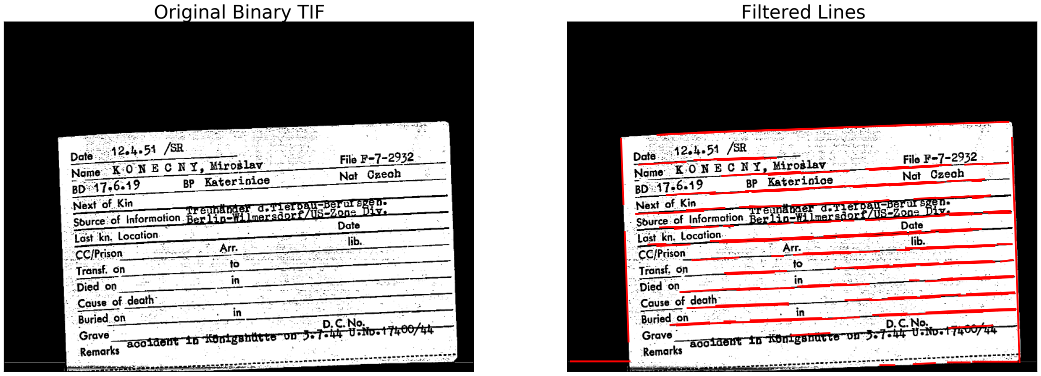
[21] CNI card of Josef Konieczny, 0.1/28005015/ITS Digital Archive, USHMM.

[7] Indeed, I have experimented with this and have found it to be an effective method of eliminating skewed false positives that were not filtered out by my edge detection algorithm, as posed in Section III. This method is robust to an overall rotation of the document in the scan image.


Original Binary TIF
Filtered Lines

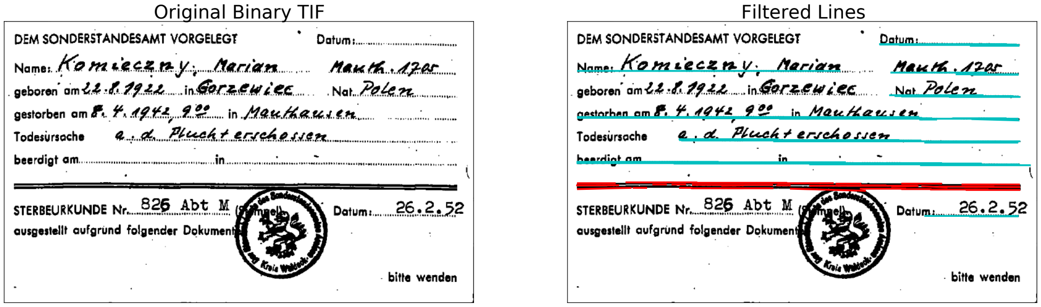
K O R N Sabina 550 506
geb. NEUBERG
Abraham u. Fryda Rochmann jüd.
15.11.1933 Leczno stl/poln.
11.40- 4.43 Gh. Warschau, Flucht
- 1.45 Illegal., befr.
1946 DPL Zeilsheim
1947 Dornstadt
3.47 ü. Frankreich, Cypern nach Israel
LA.f.Wg.Stuttg.
f.RA. Nathanson , München
sch

Fig. 2. A two panel plot showing a card with no lines on the left and the results of the line detection algorithm on the right. The line detection algorithm identifies no lines on the card, as desired. *CNI card of Sabina Korn, 0.1/28204178/ITS Digital Archive, USHMM*


Original Binary TIF
Filtered Lines

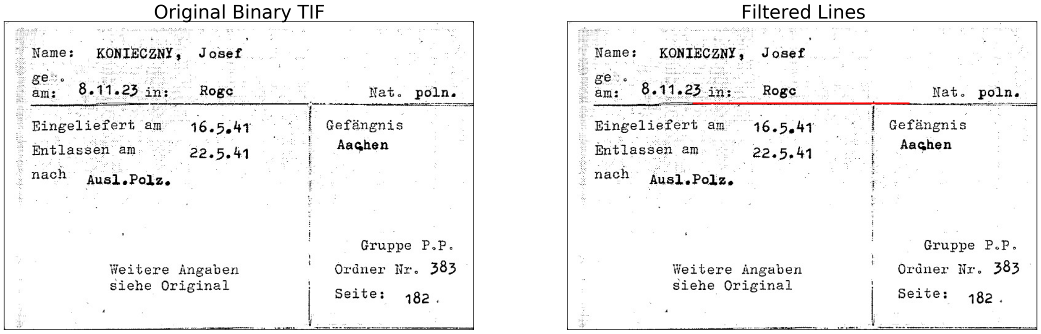
Date 12.4.51 /SR
Name K O N E C N Y, Miroslav File F-7-2932
BD 17.6.19 BP Katerinice Nat Czech
Next of Kin Treuhänder d.Tiefbau-Berufsgen.
Source of Information Berlin-Wilmersdorf/US-Zone Div.
Last kn. Location Date
CC/Prison Arr. lib.
Transf. on to
Died on in
Cause of death
Buried on in
Grave D.C.No. accident in Königshütte on 3.7.44 U.No.17400/44
Remarks

Fig. 3. A two panel plot showing a skewed card with many solid lines on the left and the results of the line detection algorithm on the right. *CNI card of Miroslav Konecny, 0.1/28003889/ITS Digital Archive, USHMM*


Original Binary TIF
Filtered Lines

DEM SONDERSTANDESAMT VORGELEGT Datum:
Name: Komieczny, Marian Mauth. 1705
geboren am 22.8.1922 in Gorzewiec Nat. Polen
gestorben am 8.4.1942, 9⁰⁰ in Mauthausen
Todesursache a. d. Flucht erschossen
beerdigt am in
STERBEURKUNDE Nr. 826 Abt M (Stempel) Datum: 26.2.52
ausgestellt aufgrund folgender Dokumente
bitte wenden

Fig. 4. A two panel plot showing a card with both solid and dotted lines on the left and the results of the line detection algorithm on the right. Solid lines are plotted in red, and dotted lines are plotted in cyan. *CNI card of Marion Komieczny, 0.1/28004164/ITS Digital Archive, USHMM*


Original Binary TIF
Filtered Lines

Name: KONIECZNY, Josef
geb.
am: 8.11.23 in: Rogc Nat. poln.
Eingeliefert am 16.5.41 Gefängnis
Entlassen am 22.5.41 Aachen
nach Ausl.Polz.
Gruppe P.P.
Weitere Angaben Ordner Nr. 383
siehe Original Seite: 182

Fig. 5. A two panel plot showing a card with solid lines on the left and the results of the line detection algorithm on the right. Here, the sensitivity of the algorithm to scan quality is evident: the vertical line is split into many shorter lines in the scan, and the algorithm fails to identify it. This failure mode is currently being explored. *CNI card of Josef Konieczny, 0.1/28005015/ITS Digital Archive, USHMM*