

Automatic removal of crossed-out handwritten text and the effect on writer verification and identification

(The original paper was published in: Proc. of Document Recognition and Retrieval XV, IS&T/SPIE International Symposium on Electronic Imaging 2008.)

Axel Brink Harro van der Klauw Lambert Schomaker

Dept. of Artificial Intelligence, University of Groningen
P.O. Box 407, 9700 AK Groningen, The Netherlands

ABSTRACT

A method is presented for automatically identifying and removing crossed-out text in off-line handwriting. It classifies connected components by simply comparing two scalar features with thresholds. The performance is quantified based on manually labeled connected components of 250 pages of a forensic dataset. 47% of connected components consisting of crossed-out text can be removed automatically while 99% of the normal text components are preserved. The influence of automatically removing crossed-out text on writer verification and identification is also quantified. This influence is not significant.

Keywords: Crossed-out text, off-line, writer verification, writer identification

1. INTRODUCTION

Computers can read constrained handwritten text reasonably well. While it still is a hard problem that has not yet been completely solved, fairly good recognizers exist.¹ These recognizers require that the text has been written in a strictly defined format. Several collections of such formatted handwritten text have been created in controlled conditions, for example see² or.³ These collections are very valuable for training automatic recognizers and for testing whether automatic recognition works in principle.

In practice, however, no assumptions about the input documents can be made. Therefore, automatic recognition of unrestricted handwritten text is still problematic. For example, when a student takes a page of notes written during a lecture, puts it in the scanner and instructs the computer to read it, it will probably produce garbage. One of the main problems is that the computer has no idea *where* the text is. Usually, everything that is dark, is seen as text. This can include elements like background print, stains, and physical damage. These elements hold no information about the handwriting and should be discarded. Since it is hard to define those elements explicitly, removing them is generally not straightforward.

Another such element is crossed-out text. Crossed-out text is intended not to be read and therefore it seems wise to identify them as such. Such elements impede automatic handwriting recognition.⁴ Identifying and discarding crossed-out text is not only relevant for text recognition, but also for writer verification and writer identification. It is often assumed that crossed-out text also impedes computation of writer specific features of the handwriting, because it is irregular.⁵ It is conceivable that feature extraction methods find many bogus features in the crossed-out text that may not be present in another text of the same writer, decreasing the apparent similarity. Therefore, it seems appropriate to attempt to remove crossed-out text prior to automatic writer verification and identification.

Identifying crossed-out text has not attracted much attention yet. One approach focused on separate characters and distinguished characters from noise including crossed-out characters.⁶ In a more recent approach,

Further author information: (Send correspondence to Axel Brink)

Axel Brink: E-mail: a.a.brink@ai.rug.nl, Telephone: +31 50 363 7410

Lambert Schomaker: E-mail: schomaker@ai.rug.nl

Harro van der Klauw: E-mail: lvdklauw@ai.rug.nl

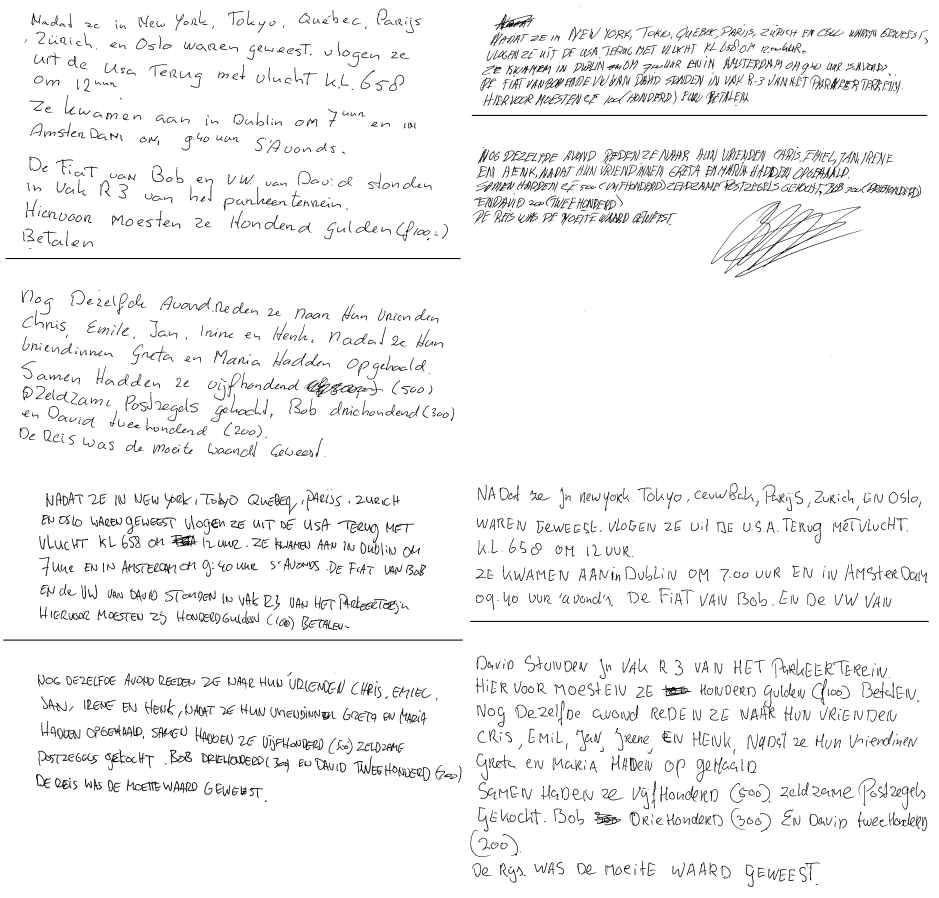


Figure 1. Example documents in the NFI dataset, each one cut in two parts.

Markov Random Fields were used to identify crossed-out words in very challenging documents.⁷ A special property of that particular method is that it seems robust against connections between crossed-out words and normal words. The results look promising, but the performance has not been quantified.

In this paper a method for identifying crossed-out words in offline handwriting is proposed. It works on the level of connected components and classifies them based on two features of the skeleton: the *branching* feature and the *size* feature. The system is trained and tested on a part of a real forensic dataset, called the NFI dataset. This dataset was first introduced in.⁵ It consists of 3500 handwritten samples taken from suspects in criminal cases; these samples have previously been studied studied manually by the NFI, the Dutch National Forensic Institute. Apart from the textual content, the handwriting is unconstrained and contains many crossed-out words. See Figure 1 for an example. This dataset seems to be somewhat similar to the kind of data used in,⁸ which consists of spontaneous handwriting.




Training and testing was performed in three stages. In the first stage, the classification performance is assessed on the level of connected components in the first 250 pages of the NFI dataset. In the second and third stage, this classification is applied to assess the effect on writer verification and identification, respectively, on 2374 pages.

2. RECOGNIZING CROSSED-OUT WORDS

2.1 Preprocessing and segmentation

The first 250 pages of the NFI dataset were thresholded using Otsu's thresholding method.⁹ From the result, the black connected components were extracted using 8-connectivity. Two kinds of components were discarded: very

Table 1. Classes of labeled connected components.

	normal	crossed-out	other
train set	43745	403	202
test set	41640	221	326
examples			

small components with a width or height smaller than 7 pixels, and very big components with a width or height bigger than half the page. The small components can be considered to be noise or dots; the big components were caused by page border effects.

This resulted in a set of 86537 connected components. These were manually labeled into three categories: “normal”, “crossed-out” and “other”. The category “other” consisted of connected components that are noise or textual elements that could not be clearly categorized into one of the other categories. This categorization proved to be not straightforward during the manual labeling process, which indicates that the problem of detection of crossed-out text may actually be ill-posed. The number of components in each class and examples are shown in Table 1.

2.2 Branching feature

When handwritten words are crossed-out, one or more strokes are written over existing strokes. The result can be seen as a high number of strokes with many crossings. The branching feature takes the number of crossings into account, where each crossing is called a branching point. To find the branching points of the connected components, they were first thinned using a recent method.¹⁰ In the resulting skeleton, the branching points were identified as the black pixels that have more than two 8-connected black neighbors. This usually results in more than one branching point per actual crossing, but this is not important for the quality of the feature. The resulting number of branching points was normalized by dividing by the width of the connected component.

2.3 Size feature

The second feature exploits the fact that crossed-out text is usually a big object, because the crossing strokes add ink and usually connect individual letters or parts of a word. The size of the object is measured by counting the number of pixels in the skeleton image. Another approach would be to count the number of pixels in the original connected component; our tests indicated that that does not make much of a difference.

2.4 Training

The just mentioned features were both normalized by dividing the feature values by the standard deviation within each page. This ensures that feature values that are not common within a page can be identified as relevant extreme values. The final result is a set of labeled two-dimensional data. This data was split in two parts: a *train* set containing the connected components from page 1–125 and a *test* set involving page 126–250. The feature values of the classes “normal” and “crossed-out” in the train set are plotted in Figure 2. The feature

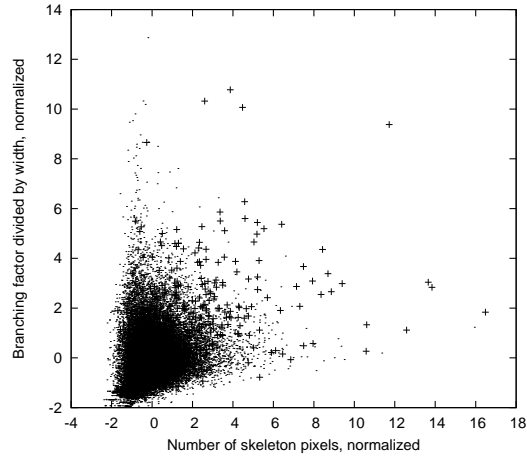
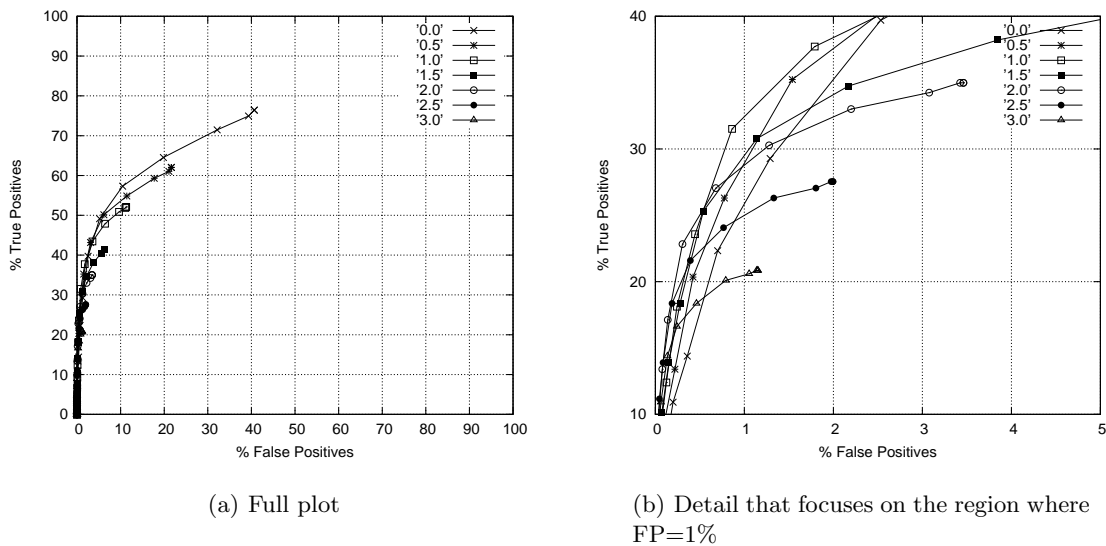


Figure 2. Feature values of connected components labeled as normal text (‘.’) and crossed-out text (‘+’) in page 1–125. Many pluses are inside the cloud of dots.



(a) Full plot

(b) Detail that focuses on the region where FP=1%

Figure 3. Multiple ROCs. Each ROC has a fixed θ_s ; see legend. Along each curve, θ_b varies. The curves do not reach the upper right corner because given the selections of θ_s , θ_b could not be positioned such that all data points would fall within the decision boundaries.

values of the class “other” were not plotted, since they are not relevant for determining a decision boundary between the features of normal words and crossed-out words.

The figure shows that the classes “normal” and “crossed-out” mainly overlap, but not totally. It also shows that there are many more instances in the “normal” class. The classes can be separated up to a certain degree by a decision tree, which is a very simple classifier. Several other classifiers have been tried as well, including k-nearest neighbor, a linear support vector machine¹¹ and a neural network, but since their performance was not better and a decision tree is simple, the latter was used.

The decision tree was implemented by setting thresholds on each of the two normalized feature values: θ_s is the threshold on the size feature; θ_b is the threshold on the branching feature. Values above both of the thresholds were seen as positive examples, or crossed-out words. By positioning the thresholds, the ratio of true positives (TP) and true negatives (TN) can be balanced. This can be done using ROC plots which are created on the train set; see figure 3 and 4.

The optimal balance between TP and TN depends on the application, but at least it is desired that most of

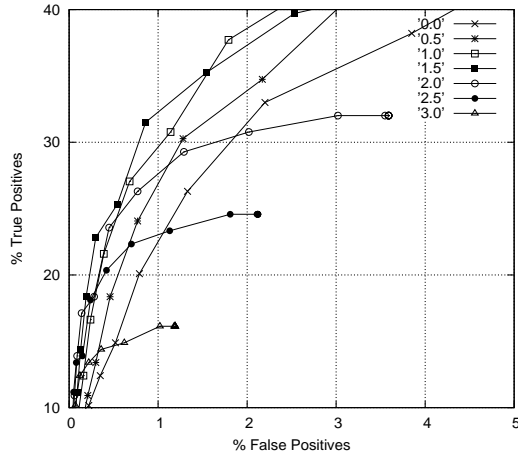


Figure 4. Same zoomed figure as Figure 3(b), but now every ROC curve has a fixed θ_b (see legend); along each ROC θ_s varies.

the normal text is not seen as crossed-out and thus remains intact. For illustrative purposes, it is now assumed that TN should be at least 99%. In other words, the number of false positives (FP) should be less than 1%. Using plots 3(b) and 4, it can be derived that usable thresholds would be $\theta_s = 1$; $\theta_b = 1.5$.

2.5 Results

The thresholds $\theta_s = 1$ and $\theta_b = 1.5$ were applied to the test set, which was completely fresh: it had not been used for training or testing before. The results are: $TP = 47.5\%$ and $TN = 99.1\%$. That means that almost half of the crossed-out words can be automatically removed while preserving 99% of the normal text. Figure 5 shows what the result would be on the images from Figure 1. Figure 6 shows the results using other thresholds. In these examples all of the crossed-out words have been successfully removed. It is clear that some of the components of normal words are removed as well, particularly bigger components, but most components of normal words remain.

To illustrate how the method scales to very big scratches, a small experiment was also performed on semi-artificial data: the four pages of which the cutted versions are shown in Figure 1 have been overlaid with pages containing big scratches. For this test the condition that the crossed-out components should not be bigger than half the page was relaxed. Figure 7 shows what the result would be on such pages.

3. APPLICATION TO WRITER VERIFICATION

The proposed technique to automatically remove crossed-out words was applied in a writer verification experiment to determine whether it affects performance. Writer verification means that a decision must be made whether two documents have been written by the same person. This can be done automatically by first computing writer-specific feature vectors of the input documents and then applying a threshold on a distance measure between the feature vectors. In this experiment, the powerful *Hinge*¹² feature was used. This technique captures the orientation and curvature of the ink trace, encoded in a 528-dimensional feature vector. As a distance measure, the χ^2 measure¹³ was used. It is defined (after renaming) as:

$$d_{\chi^2}(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{|\mathbf{a}|} \frac{(\mathbf{a}_i - \mathbf{b}_i)^2}{\mathbf{a}_i + \mathbf{b}_i}$$

where i is an index to the elements of feature vectors \mathbf{a} and \mathbf{b} . Documents with a χ^2 distance below a certain threshold were classified as belonging to the same writer.

The experiment was performed as follows. First, like in,⁵ all pages of the NFI dataset were split in an upper part and a lower part. 1127 pages had to be discarded because curved or sloped baselines made a good cut

a n ze in New oak, Tok o, Qu'bec, Parijs
 Zürich, en Oslo waren geweest. v ogen ze
 uit de Usa Terug v vlucht kl. 65
 om 12 uur
 Ze waren aan in Dubin o 7 uur en
 Ams Da i om 40 uur S'Avon s.
 iet va b eg vW van au: ston n
 in Vak R3 van h j par n in.
 Hi voo o s en ze l on n gul n i, i)
 Betalen

Nog Dezelfde Avond. Reden ze n n an vrienden
 Chris, Emil, Jan, l'ne en Henk. Na ad ze Han
 bri n innen Greta n lara Ha n opg led.
 Samen adden 2 vijf onden i (500)
 Zeldzame Postzegels l Bob d'chon end (300)
 en Davi d'uekon nel (200).
 De Reis was mo' laand Gev l.

N AT ZE IN NEW YORK, TOKYO, QUEBEC, PARIS, ZÜRICH
 EN OSLO WAREN GEWEEST. VLOGEN ZE UIT DE USA TERUG MET
 VLUCHT KL. 65 OM 12 UUR. ZE WAREN AAN IN DUBOIN
 OM 7 UUR EN IN AMSTERDAM OM 40 UUR 'AVOND.
 HIER MOESTEN ZE WAARD GEWEEST.

Nog Dezelfde Avond - Ze n r huu v'leene c is, ethec,
 s, l me en enk, d t ze huu v'ne inni Greta en j
 Haq op- . s - o H en ze v'p'han d (500) -
 Postzegels o t . E (300) en i Twee n (200)
 De Reis was mo' laand Gev l.

D'v' d' stou n in v k R3 v n et par eek'ter'inv.
 ier voor moest'ev ze oude n v'len 100 l' em.
 Nog Dezelfde avond Reden ze n l hun vrienden
 Chris, Emil, Jan, j e 'n enk, n d'el ze un v'ne inni
 kete en Marki H en o Magid
 Men goen v'g on RD (500) zel zame Pos zegels
 g'ekocht. Bob rie onde n (300) en Davi ee . j
 (200).
 De Reis was mo' laand Gev l.

Figure 5. Images from Figure 1; thresholds $\theta_s = 1$ and $\theta_b = 1.5$ applied. All crossed-out components have been removed at the expense of some normal text.

Table 2. Number of pages and writers in the dataset for verification and identification.

	train set	test set
original pages (writers)	250	3250
selected pages (writers)	181 (87)	2193 (988)
selected parts (writers)	362 (87)	4386 (988)

impossible. After splitting, 4748 page parts remained, written by 1074 persons. The page parts were converted to monochrome using Otsu thresholding because this is required by the method to remove crossed-out words. The page parts were divided into two sets: a train set and a test set. The train set consisted of the parts of pages 1–250; the test set consisted of the parts from the other 3250 pages. See Table 2 for details. To compute the baseline performance, a verification threshold was learned from the training data by modeling the distances in the “same writer” and “different writer” classes using Parzen windowing. The threshold was selected such that the expected ratio of true positives (TP) was equal to the expected ratio of true negatives (TN); the equal-error rate (EER). This threshold was applied to the test set, yielding the experimental TP and TN.

To assess the effect of crossed-out text, the same steps were taken on the same page parts after cleaning by the crossed-out text removal method. Several values of θ_s and θ_b were tried while testing on the train set. The results are shown in Table 3. The table shows that the values for θ_s and θ_b have no big implications on writer verification, but the best result was used to determine the final values: $\theta_s = 2$ and $\theta_b = 2$.

naat ze in New York, Tokyo, Quebec, Parijs, Zürich, en Oslo waren geweest. vlogen ze uit de Usa terug met vlucht kl. 650 om 12 uur.
Ze kwamen aan in Dublin om 7 uur en in Amsterdam om 9:40 uur 's Avonds.

De Fiat van Bob en VW van David stonden in Vak R3 van het parkeerterrin.
Hiervoor moesten ze honderd gulden betalen.

Nog dezelfde avond reden ze naar hun vrienden Chris, Emile, Jan, Inne en Henk. Nadat ze hun vriendinnen Greta en Maria hadden opgehaald, samen adden 2 vijfhonderd (500) zeldzame postzegels gekocht, Bob driehonderd (300) en David tweehonderd (200).
De reis was de moeite waard geweest.

NAAT ZE IN NEW YORK, TOKYO, QUEBEC, PARIJS, ZÜRICH EN OSLO WAREN GEWEEST. VLOGEN ZE UIT DE USA TERUG MET VLUCHT KL. 650 OM 12 UUR. ZE KWAMEN AAN IN DUBLIN OM 7 UUR EN IN AMSTERDAM OM 9:40 UUR 'S AVONDS. DE FIAT VAN BOB EN DE VW VAN DAVID STONDEN IN VAK R3 VAN HET PARKEERTERRIN. HIERVOOR MOESTEN ZE HONDERD GULDEN (1) BETALEN.

NOG DEZELFDE AVOND. ZE WAREN HUN VRIENDEN CHRIS, EMILE, JAN, INNE EN HENK, NA DAT ZE HUN VRIENDINNE GRETA EN MARIA HADEN OPGEHAALD. SAMEN HADEN ZE VIJFHONDERD (500) ZELDZAME POSTZEGELS GEKocht. BOB DRIEHONDERD (300) EN DAVID TWEEHONDERD (200).
DE REIS WAS DE MIDDLE WAARD GEWEEST.

NAAT ZE IN NEW YORK, TOKYO, QUEBEC, PARIJS, ZÜRICH EN OSLO WAREN GEWEEST. VLOGEN ZE UIT DE USA TERUG MET VLUCHT KL. 650 OM 12 UUR. ZE KWAMEN AAN IN DUBLIN OM 7 UUR EN IN AMSTERDAM OM 9:40 UUR 'S AVONDS. DE FIAT VAN BOB EN DE VW VAN DAVID STONDEN IN VAK R3 VAN HET PARKEERTERRIN. HIERVOOR MOESTEN ZE HONDERD GULDEN (1) BETALEN.

NOG DEZELFDE AVOND. ZE WAREN HUN VRIENDEN CHRIS, EMILE, JAN, INNE EN HENK, NA DAT ZE HUN VRIENDINNE GRETA EN MARIA HADEN OPGEHAALD. SAMEN HADEN ZE VIJFHONDERD (500) ZELDZAME POSTZEGELS GEKocht. BOB DRIEHONDERD (300) EN DAVID TWEEHONDERD (200).
DE REIS WAS DE MIDDLE WAARD GEWEEST.

NA DAT ZE IN NEW YORK, TOKYO, QUEBEC, PARIJS, ZÜRICH, EN OSLO WAREN GEWEEST. VLOGEN ZE UIT DE USA TERUG MET VLUCHT KL. 650 OM 12 UUR. ZE KWAMEN AAN IN DUBLIN OM 7:00 UUR EN IN AMSTERDAM OM 9:40 UUR 'S AVONDS. DE FIAT VAN BOB EN DE VW VAN DAVID STONDEN IN VAK R3 VAN HET PARKEERTERRIN. HIERVOOR MOESTEN ZE HONDERD GULDEN (100) BETALEN.

DAVID STONDEN IN VAK R3 VAN HET PARKEERTERRIN. HIERVOOR MOESTEN ZE HONDERD GULDEN (100) BETALEN. NOG DEZELFDE AVOND. ZE WAREN HUN VRIENDEN CHRIS, EMILE, JAN, INNE EN HENK, NA DAT ZE HUN VRIENDINNE GRETA EN MARIA HADEN OPGEHAALD. SAMEN HADEN ZE VIJFHONDERD (500) ZELDZAME POSTZEGELS GEKocht. BOB DRIEHONDERD (300) EN DAVID TWEEHONDERD (200).
DE REIS WAS DE MIDDLE WAARD GEWEEST.

Figure 6. Images from Figure 1; thresholds $\theta_s = 2.5$ and $\theta_b = 2.5$ applied. With less strict thresholds, more of the normal text remains. Crossed-out text is also more likely to remain, but that does not occur in this example.

Table 3. Writer verification results on 362 thresholded half pages extracted from the train set (the first 250 pages) of the NFI dataset; 87 writers.

θ_s	θ_b	TP	TN
1	1	78.1%	81.9%
1	1.5	79.3%	81.5%
1.5	1	78.7%	81.3%
1.5	1.5	79.7%	80.7%
1.5	2	80.4%	80.3%
2	1.5	80.3%	80.5%
2	2	79.0%	82.8%
2.5	2.5	79.8%	81.3%
3	3	80.5%	81.1%

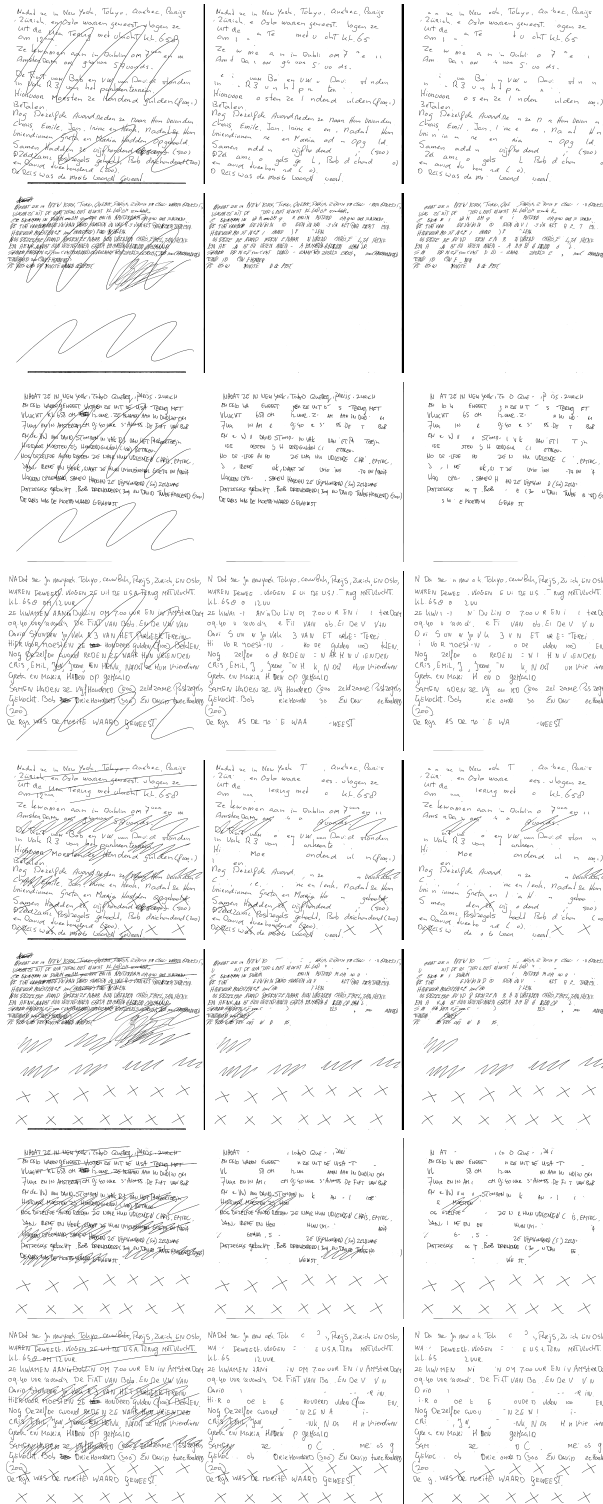


Figure 7. Result of removing big scratched-out elements. Column 1: original overlaid with scratches, column 2: result with $\theta_s = 2.5$, $\theta_b = 2.5$, column 3: result with $\theta_s = 1.0$, $\theta_b = 1.5$.

Table 4. Verification results on 4386 thresholded half pages; 988 writers.

	TP	TN
Baseline	76.6%	84.0%
$\theta_s = 2, \theta_b = 2$	77.1%	83.6%

Table 5. Identification results on 362 thresholded half pages extracted from the first 250 pages of NFI dataset; 87 writers.

θ_s	θ_b	Top-1	Top-10	Top-100
1	1	85.4%	95.6%	98.9%
1	1.5	87.9%	96.1%	99.2%
1.5	1	86.7%	95.0%	99.2%
1.5	1.5	87.3%	96.1%	99.2%
1.5	2	88.4%	95.0%	99.2%
2	1.5	87.0%	96.1%	99.2%
2	2	88.1%	95.6%	99.2%
2.5	2.5	88.1%	95.0%	98.9%
3	3	87.9%	95.0%	99.2%

3.1 Results

The values $\theta_s = 2$ and $\theta_b = 2$ were used to remove crossed-out text in the test set (4386 half pages) of the NFI dataset. Table 4 shows the result together with the baseline performance. It is clear that automatically removing crossed-out text using the proposed method has no substantial influence on writer verification performance.

4. APPLICATION TO WRITER IDENTIFICATION

The same kind of experiment was performed to determine the effect of automatically removing crossed-out text on writer identification. Writer identification means returning a *hit list*, a sorted list of documents of which the handwriting is similar to that of a questioned document. In this experiment, similarity was again determined by the hinge feature and χ^2 -distance. The pages of the NFI dataset were split in parts, thresholded, and divided into a train set and a test set as described in section 3. The baseline performance was computed by treating every document in the test set as a questioned document, then yielding the hit list and finally counting how often a matching document appeared in the top-1, top-10 or top-100.

This was also done with pages of which crossed-out text was automatically removed using several values of θ_s and θ_b in the train set. The performance using these thresholds on the train set is shown in Table 5. Although the differences are again very small, the best selection of thresholds could be identified: $\theta_s = 1.5$ and $\theta_b = 2$.

4.1 Results

The thresholds $\theta_s = 1.5$ and $\theta_b = 2$ were applied to remove crossed-out text in the test set. On the resulting documents, writer identification was performed. The results are shown in Table 6, together with the baseline performance. The table shows that automatically removing crossed-out text does not improve writer identification performance.

5. CONCLUSION

In this paper a simple method to identify and remove crossed-out text was presented. It can remove 47% of crossed-out text while 99% of the normal text is preserved. There is no important effect on writer verification or identification based on the hinge feature.¹² This is an indication that the effect of crossed-out text on writer verification and identification may be overestimated.

Table 6. Identification results on 4386 thresholded half pages; 988 writers.

	Top-1	Top-10	Top-100
Baseline	76.5%	88.1%	95.0%
$\theta_s = 1.5, \theta_b = 2$	75.5%	87.7%	94.8%

6. DISCUSSION

Although our result suggests that removing moderate crossed-out text may not be worth the effort, there are options to make this statement more firm. It is conceivable that the Hinge feature that was used for the writer verification and identification experiment, is just quite robust for crossed-out text. Therefore, other features should be tried for this too, for example the Fraglets feature (also called fCO3).¹⁴ It is also possible that the automatic method to remove crossed-out words does improve verification or identification performance, but at the same time reduces performance because also some good text is removed. Therefore the next step should be to improve the method to detect crossed-out words. One way to improve the method could be to use textural features such as Hinge on the level of connected components. Alternate thinning methods could be tried for the branching feature because artefacts in the skeleton have a big influence on the performance. The method could also be adapted to work with grayscale images, which would make the method more versatile. That would slightly improve writer verification and identification performance, since the hinge feature was designed for grayscale and performs a bit worse on black and white images. The best values for θ_s and θ_b could be determined in a more thorough way by using steepest descent or genetic algorithms. Furthermore, other classifiers can be tried, and line segmentation should be applied to disconnect connected components that are big because they consist of intersecting text from multiple text lines.

A final question that remains unanswered for now is how much text can be crossed-out without disturbing automatic verification or identification. It is hard to imagine that everything could be crossed-out for free, so we wonder how far one can go.

REFERENCES

1. R. Plamondon, S. Srihari, E. Polytech, and Q. Montreal, "Online and off-line handwriting recognition: a comprehensive survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **22**(1), pp. 63–84, 2000.
2. U. Marti and H. Bunke, "A full english sentence database for off-line handwriting recognition," in *Proc. of the 5th ICDAR*, pp. 705–708, 1999.
3. L. Schomaker and L. Vuurpijl, "Forensic writer identification: A benchmark data set and a comparison of two systems," tech. rep., NICI, Nijmegen, 2000.
4. E. Lecolinet, L. Likforman-Sulem, L. Robert, F. Role, and J. Lebrave, "An integrated reading and editing environment for scholarly research on literary works and their handwritten sources," *Proceedings of the third ACM conference on Digital libraries*, pp. 144–151, 1998.
5. A. Brink, L. Schomaker, and M. Bulacu, "Towards explainable writer verification and identification using vantage writers," in *ICDAR*, pp. 824–828, 2007.
6. J. Arlandis, J. C. Perez-Cortes, and J. Cano, "Rejection strategies and confidence measures for a k-nn classifier in an ocr task," in *ICPR*, 2002.
7. S. Nicolas, T. Paquet, and L. Heutte, "Markov random field models to extract the layout of complex handwritten documents," in *Proc. of the 10th IWFHR*, 2006.
8. A. H. Toselli, A. Juan, and E. Vidal, "Spontaneous handwriting recognition and classification," in *ICPR*, pp. 433–436, 2004.
9. N. Otsu, "A threshold selection method from gray-level histograms," *IEEE trans. on Systems, Man and Cybernetics* **9**(1), pp. 62–66, 1979.
10. L. Huang, G. Wan, and C. Liu, "An improved parallel thinning algorithm," in *ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition*, p. 780, IEEE Computer Society, (Washington, DC, USA), 2003.
11. T. Joachims, "Making large-scale svm learning practical," in *Advances in Kernel Methods - Support Vector Learning*, B. Schölkopf, C. Burges, and A. Smola, eds., MIT-Press, 1999.
12. M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)* **29**(4), pp. 701–717, 2007.

13. L. Schomaker and M. Bulacu, "Automatic writer identification using connected-component contours and edge-based features of uppercase western script," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(6), pp. 787–798, 2004.
14. L. Schomaker, M. Bulacu, and K. Franke, "Automatic writer identification using fragmented connected-component contours," in *9th IWFHR*, F. Kimura and H. Fujisawa, eds., pp. 185–190, (Tokyo, Japan), October 2004.