An Image Processing Self-Training System for Ruling Line Removal Algorithms

Konstantinos Prokopiou[1]
[1]Hellenic Open University
Patras, Greece
k11sp8u@gmail.com

Ergina Kavallieratou[1,2]

Efstathios Stamatatos[2]
[2]University of the Aegean
Samos,Greece
{kavallieratou,stamatatos}@aegean.gr

***Abstract*— Ruling line removal is an important pre-processing step in document image processing. Several algorithms have been proposed for this task. However, it is important to be able to take full advantage of the existing algorithms by adapting them to the specific properties of a document image collection. In this paper, a system is presented, appropriate for fine-tuning the parameters of ruling line removal algorithms or appropriately adapt them to a specific document image collection, in order to improve the results. The application of our method to an existed line removal algorithms is presented.**

Keywords—Document Image Processing; Ruling Line Removal; Simulated Annealing

I. INTRODUCTION

The presence of ruling lines is very common in document images in order to help the writer to keep certain rules e.g. text in order, text inside specific area etc. However, the ruling lines make more difficult several procedures, like document image processing or OCR. Thus, a pre-processing task dealing with ruling line detection and removal is required in many cases.

In more detail, the existence of ruling lines on a binarized manuscript can cause several problems in document image analysis:

- some common techniques like the extraction of connected components may not be applied correctly due to the overlap of ruling lines with the characters
- certain basic processing tasks including text extraction, page segmentation, line segmentation, character recognition, etc. are considerably harder.

On the other hand, the thickness of ruling lines may vary, even in the same page (e.g., 1-2 pixels), while some lines may be broken or skewed. Therefore, their detection and removal is complicated, given that in binarized document images there is no color discrimination between ruling lines and text. Moreover, an attempt to remove ruling lines could introduce more noise since some parts of the characters will also be removed or some parts of the line could be left behind. In both cases, the precision of page segmentation and document recognition tasks will be reduced.

A variety of algorithms for ruling line detection and removal have been proposed [1-3]. Many of the suggested algorithms concern specific type of document form [4-5], while others can fail in the presence of broken or skewed lines or in the case that text and ruling lines are extremely overlapped [6]. In the last case, a decision has to be made if a black pixel belongs to a line or a character, in order to turn it off or keep it on, respectively. Some techniques decide based on the line characteristics and the character contour analysis [7-8], while others intend to repair the characters, after the line removal [9]. Moreover, in many cases several parameters of the algorithms need to be appropriately tuned in order to increase the effectiveness. Thus, the problem of ruling line detection and removal practically remains open as far it concerns the effectiveness over any type of document image, in general.

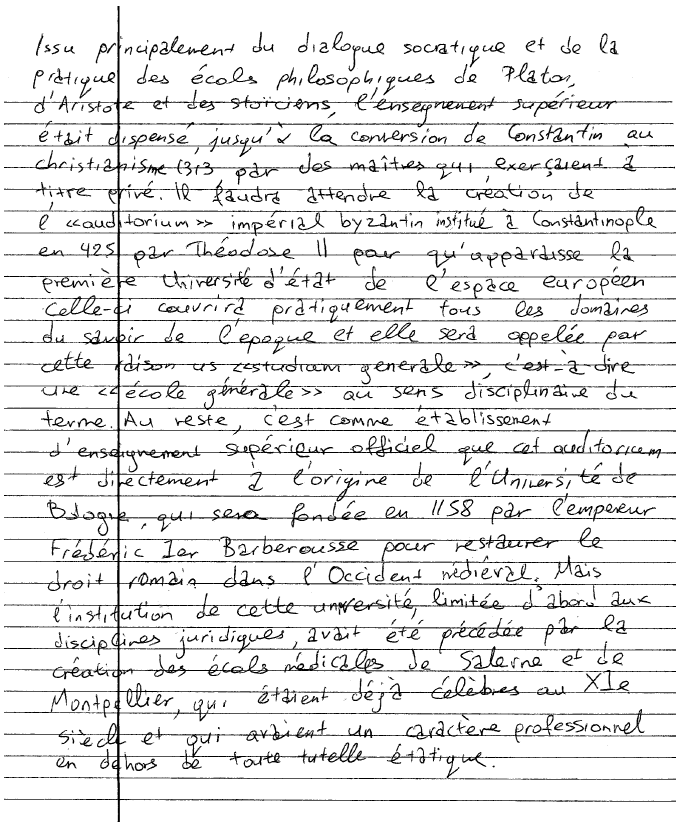
Issu principalement du dialogue socratique et de la pratique des écoles philosophiques de Platon, d'Aristote et des stoïciens, l'enseignement supérieur était dispensé, jusqu'à la conversion de Constantin au christianisme (313, par des maîtres qui exerçaient à titre privé. Il faudra attendre la création de l'«auditorium» impérial byzantin institué à Constantinople en 425, par Théodose II pour qu'apparaisse la première université d'état de l'espace européen celle-ci couvrira pratiquement tous les domaines du savoir de l'époque et elle sera appelée par cette raison «studium generale», c'est-à-dire une «école générale» au sens disciplinaire du terme. Au reste, c'est comme établissement d'enseignement supérieur officiel que cet auditorium est directement à l'origine de l'Université de Bologne, qui sera fondée en 1158 par l'empereur Frédéric Ier Barberousse pour restaurer le droit romain dans l'Occident médiéval. Mais l'institution de cette université, limitée d'abord aux disciplines juridiques, avait été précédée par la création des écoles médicales de Salerne et de Montpellier, qui étaient déjà célèbres au XIe siècle et qui avaient un caractère professionnel en dehors de toute tutelle étatique.

Figure 1. A synthetic document image with ruling lines.

Another crucial aspect is the objective evaluation of the proposed algorithms. Tagged ground truth data, including document images of any type, where each pixel would be

tagged according to whether it should be removed or not, do not exist. Due to this lack, in many research papers, the proposed algorithms are evaluated based on synthetic data [10-11], where ruling lines are artificially introduced in existing documents. Although the synthetic document images mostly lack the natural alignment of ruling lines with the text, they provide a more general form of documents where the writer do not necessarily respect the presence of ruling lines and the lines intersect with the text (Fig. 1).

In this paper, a new system is proposed attempting to improve the application of ruling line detection and removal algorithms. The system implements a technique for the automatic training of a line removal. In other words, the parameters of an existing algorithm are fine-tuned based on the characteristics of a document collection. To this end, the Simulated Annealing algorithm is used to estimate the most appropriate parameter values. In order to demonstrate the proposed technique, we applied it to an existed algorithm of ruling line removal.

In order to choose an algorithm for application, we posed several criteria:

- The algorithms should be parameterized in order to be adapted to document collections.
- They should be appropriate for binarized images.
- They should be able to handle several typical problems: broken lines, skewed pages, overlapping with text, etc.
- The resulted text should remain in good condition.

The contribution of the proposed system consists of the automatic estimation of the appropriate parameter values of an algorithm, given very few annotated pages. Our method can estimate the appropriate parameter values for a document collection. To the best of our knowledge, this is the first time that such a methodology is developed for the automatic training of the line removal algorithms.

The rest of this paper is organized as follows: in section II, a short review of the state-of-the-art in ruling line detection and removal is given, including a more detailed description of the algorithm that is going to be used for the demonstration of our system. In section III, the Simulated Annealing algorithm is briefly described, as well as the proposed system. Then, in section IV the appliance of our system is included. Finally, the conclusions drawn by this study are given in section V.

II. STATE OF THE ART

Several methodologies have been proposed in order to deal with the existence of ruling lines in document images. The projection profiles were used by Arvind et al. [9], where the peaks of the horizontal projection profile are considered as an estimation of the horizontal ruling lines. Next, the run-length of the line, if exceeds a threshold, will give more details. In a similar way, Cao et al. [12] first segment the page in vertical areas and then they calculate the horizontal projection profiles of each area. Hough transform has been used mostly for line and frame detection in forms [13]. The morphological operators have also been used in line removal [14-15]. However, Ye et al. [16] claim that they are not appropriate in the case of mixed text (printed and handwritten), when there is overlapping between text and lines or the thickness of the line varies. Almageed et al. [11] use a linear subspace model and decides if a pixel belongs to a ruling line or text by examining a feature vector that describes the characteristics of its neighborhood.

Shi and Govindaraju [17] and Shi et al. [18] use the fuzzy run-length for each black pixel, towards both directions, in order to estimate the existence of ruling lines. They allow breaks of white pixels in the line. Shi et al. [18] (Fig.2) described a robust and parameterized methodology that takes into account the coherence between black pixels of the same text line, while on the other hand allows the existence of white pixels, useful in the case of broken lines. It makes use of the fuzzy runlength [17].

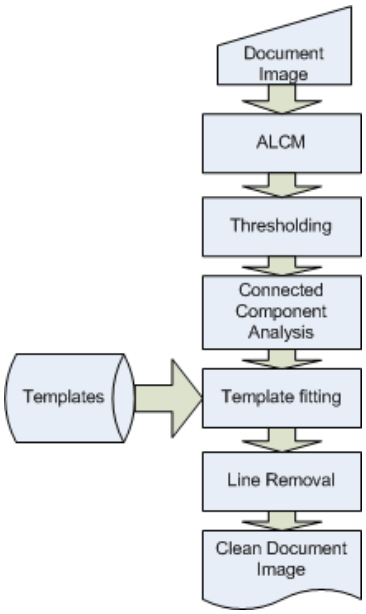

Figure 2. The first selected algorithm.

For the ruling line detection an adaptive local connectivity map (ALCM) is created, that describes the pixel coherence by using the fuzzy runlength. The fuzziness is expressed by the variety in the amount of consecutive black pixels that can be met towards both, left and right, sides. This depends on the pixel neighborhood. Given that ruling lines can be broken, an important parameter is the amount of white pixels that is accepted in order to characterize an amount of consecutive black pixels as ruling line. To emphasize the pixels of the ruling lines, the authors used the binarization algorithm of Giuliano et al. [19]. This algorithm uses 5 parameters $K_1,\ldots, K_5$. In our implementation, the authors' suggestion was followed and it was set $K_5=K_2$, while K_4 was discarded. Thus, the parameters K_1, K_2, K_3 were also used in the parameterization procedure of our system. More specifically, this algorithm concerns for each pixel two square areas: one of size A_1 around the pixel and four of size A_2 located diagonally to A_1. Both sizes, A_1 and A_2 are even. Thus, for every pixel a value *R is* calculated as:

$$R = K_1\left(\frac{1}{mesA_1}\left[\sum_{A_1}^{y}\sum_{A_1}^{x} S(x,y)\right] - \frac{K_3}{mesA_2'}\left[\sum_{A_2'}^{y}\sum_{A_2'}^{x} S(x,y)\right]\right) \quad (1)$$

where $mesA_1$ and $mesA_2$ are the amount of pixels in the corresponding areas and $S(x,y)$ the value of the corresponding pixel in ALCM. The pixel is considered black if $R > 0$ and $S(x,y) \geq K_2 / K_1$.

Next the ruling line removal procedure follows. A connected component analysis technique is applied to the image (Davies [20]) in order to collect possible line patterns. For every pattern, the thickness of the line in every pixel column is kept as well as the mean thickness. By considering the areas with thickness less or equal to the mean thickness, a best fitting line, using the linear regression method, is estimated. Then, the ruling line is re-constructed using the estimated best fitting line. Using the line patterns, the black pixels that fit these patterns are turned into white pixels.

In this technique the parameters in use (Fig.3) are:

- SBP: the accepted amount of white pixels
- W: determines the size of the squared areas A_1 and A_2, as $2*W+1$
- K_1, K_2 and K_3 : binarization algorithm parameters.

Figure 3. The parameters and their domains for the algorithm.

III. THE PROPOSED SYSTEM

A. Simulated Annealing

The Simulated Annealing (SA) is a heuristic search algorithm. It is inspired from the metallurgy [21], as a methodology that handles the heating and freezing of a material in order to increase the size of the crystals and decrease their imperfections. The heating pushes the atoms of the material off their initial state (local minimum energy) to higher energetically states. By the controlled cooling the atoms pass to states of lower energy. Kirkpatrick et al. [22] and Černý [23] were inspired from this procedure and built a search algorithm.

Finding the most appropriate parameter values of a ruling line removal algorithm, it can be seen as searching in the space defined by the possible parameter values. The application of SA to the tuning of the parameters for the algorithms is considered here. The system starts from an initial state with higher T_0 (corresponds to the initial temperature). The procedure is repeated as many times is determined by the parameter TIME and looks for the state with the minimal energy. In each repetition, the parameter T is calculated by the formula proposed by Press et al.[24]:

$$T = T_0 \cdot [1 - (t / TIME)]^{\alpha} \quad (2)$$

where $\alpha = 2$, $TIME = 45$ και $T_0 = 100$. The parameter T represents the value of the current repetition t. If this energy is lower than the previous one, the state changes. If it is not, then a probability is calculated by $(T/T_0) \cdot e^{-\Delta E/T}$, where ΔE is the energetic difference between the two states. If this probability is bigger than the predefined possibility P, the state changes, otherwise not.

In our case, energy is considered the opposite to the F-measure, in order to keep on looking for the minimum energy. Thus:

1. The algorithm is applied to an image for a certain set of parameter values,
2. The result is evaluated based on the ground truth image
3. The F-measure (energy) is calculated
4. Simulated annealing is applied and the parameter values are changing appropriately.
5. The procedure is repeated for times, TIMES, by step 1.

The random determination of P enables the search algorithm to avoid local minima.

B. System Description

Our system is appropriate to train ruling line detection and removal techniques by an automatic configuration of their parameters. The system is appropriate to check in detail the performance of an algorithm on a specific image. It can tune its parameters based on a specific document image collection by using the Simulated Annealing (section III.A).

Probably, the computer could try all the possible combinations of parameters, apply to the specific image and choose the best combination by the results. However, such an approach could result large demands in computational space and time. An algorithm of M parameters, that could take N different values each, should run N^M times. Moreover, there are continuous parameters, i.e. infinite values.

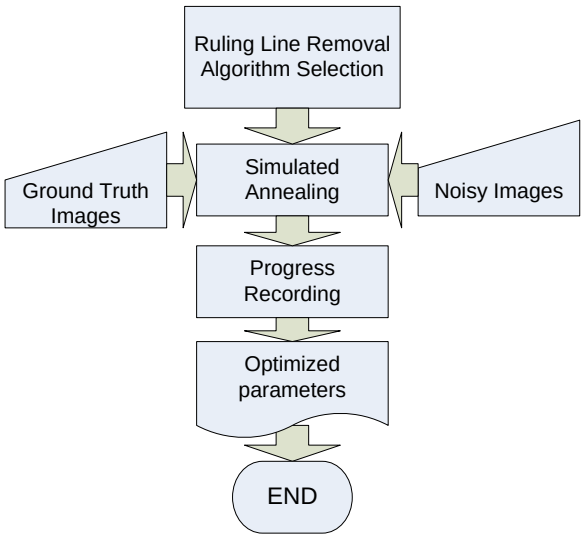

Figure 4. Parameter configuration by using Simulated Annealing.

In our system, the user can select the image that wishes to process. He also selects the ruling line detection and removal algorithm he wishes to use. Moreover the user can change the values of the parameters, if he does not agree with the by-default values that appear in the parameter section.

Once a Ruling Line Detection and Removal algorithm is selected, the system is aware of its parameters to-be-tuned, defined during the algorithm insertion procedure. The Simulated Annealing executes several repetitions in order to try multiple combinations of the parameters and select the best (fig.4). The user can watch the progress in each parameter value, as well as all the tested combinations. The best combination is finally chosen defining the tuned parameters of the algorithm.

The proposed system also provides the possibility to apply the tuned algorithm to a batch of files. This is very useful; since similar problems can be present in the same document collection. This gives the possibility to train an algorithm for one or more documents from a specific collection and apply it to the whole collection.

IV. Experimental Results

In this section we describe the database that was used for the experiments, the metrics that were used for the evaluation, as well as the experiments that were performed. Our experiments were realized in a computer with CPU Pentium(R) Dual-Core 2.50GHz.

The evaluation methodology described in **[**11**]** was used. In our database, 10 scanned page images with ruling lines from different note pads were used and combined with 10 images of text from different languages written by different persons (3 English pages, 2 Greek, 2 German, 1 French and 2 Arabic), resulting 100 images of text with ruling lines. In the present system, 3 of these images were used for training the parameters by applying Simulated Appealing, while all the rest were used to test the estimated parameters.

The images 2, 49 and 93 of our database were used to configure the algorithm parameters. These images present certain differences, in the ruling lines: Image 2 (fig.5) has almost perfect ruling lines and very little text, image 49 (fig.6) includes very light ruling lines and a lot of text, while image 93 (fig.7) presents middle-apparent ruling lines and thick text..

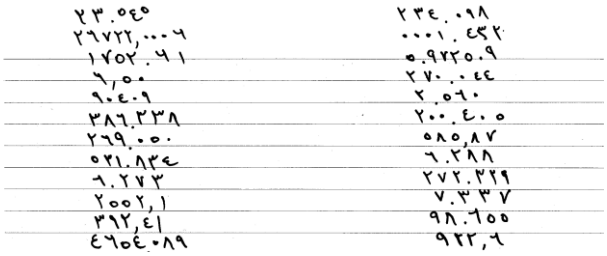

Figure 5. Synthetic Image 2.

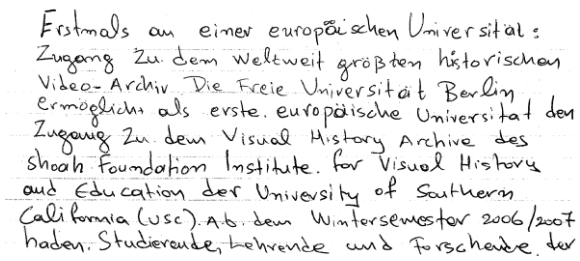

Figure 6. Synthetic Image 49

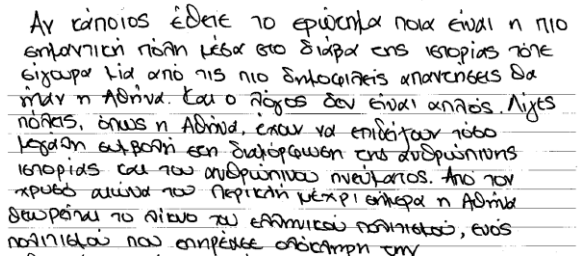

Figure 7. Synthetic Image 93.

The evaluation, as in [11], is performed via recall, precision and weighted harmonic mean F1 metrics, defined as:

$$precision = \frac{tp}{tp + fp} \tag{3}$$

$$recall = \frac{tp}{tp + fn} \tag{4}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

Where:

- True positive pixel (*tp*): a pixel that exists in both the detection map (pixels to be removed) and the rule line ground truth map (all the ruling line pixels), but not in the text only map.
- False positive pixel (*fp*): a pixel that exists in both the detection map and the text only map.
- False negative pixel (*fn*): a pixel that exists in the ground truth map, but in neither the detection map or the text only map.

The tuned parameters from the application of the proposed methodology to the algorithm [17-18], is shown in table 1, while in table 2 the evaluation metrics are presented after the application to the training images.

Table 1. Estimated parameters by Simulated Annealing.

Param.	**SBP**	**W**	**K_1**	**K_2**	**K_3**
Values	2	0	7.47	4.50	5.35

Table 2. Experim. results for the parameter values of table 1.

Image	Precision	Recall	F1-meas.
2	0.9998	0.9943	0.9970
49	0.8053	0.9497	0.8715
93	0.9743	0.9591	0.9667
Average	**0.9265**	**0.9677**	**0.9451**

Next, the algorithm with the estimated parameters were applied to the rest of the database and the mean values for the evaluation metrics were precision=0.9209, recall=0.8121 και F_1=0.8365.

For the specific dataset the results are comparative to the ones presented in Kavallieratou et al. [25], where precision=0.81, recall=0.92 and F1=0.86. Shi et al. [14] used a different dataset and evaluation methodology, counting that 95% of the pixels, belonging to ruling lines, where correctly removed.

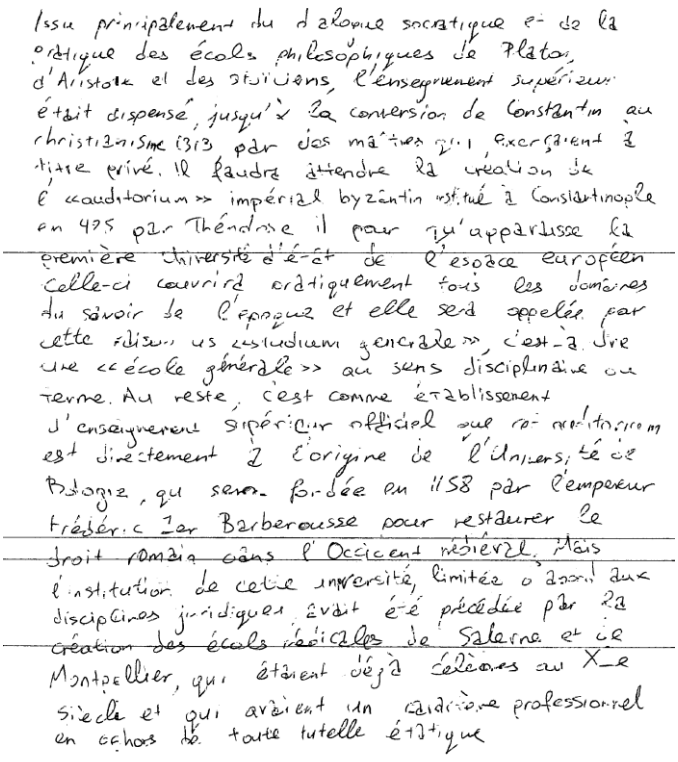

Figure 8. The result from the application of the proposed system to image of fig.1.

Although the results are quite satisfactory, several images were noticed with low values. Choosing one of them (Image 5 presenting precision=0.9999, recall=0.0860 και F_1=0.1584) and re-applying Simulated Annealing, the parameters changed to the ones shown in table 3.

Table 3. Estimated parameters by Simulated Annealing for Image 5.

Parameters	SBP	W	K_1	K_2	K_3
Values	7	5	2.66	0.24	3.68

The evaluation metrics applying these values to the specific image were precision=0.9992, recall=0.9429 και F_1=0.9702, while on the whole database precision=0.6503, recall=0.6128 και F1=0.5943. Finally, keeping the best result for each image: precision=0.9184, recall=0.8828 και F1=0.8927.

It seems that the parameter W is crucial in the case that the ruling lines on the same page present large variation in thickness, e.g. in our dataset, when there are vertical ruling lines in page (image 5). Such is the case of the image of Fig.1. Please check the result in Fig.8.

V. CONCLUSION

In this paper, a system was built, appropriate to train ruling line removal algorithms or/and adapt them to document image collections. A ruling line removal algorithm was implemented and included to our system in order to test it. In order to train this, only three images from a synthetic database of 100 images were used and the whole database was used for the evaluation.

New algorithms can easily introduced to the system, while any document image can be used as training image in order to adapt an algorithm to a specific document image collection.

The proposed system is addressed to users of document analysis software that wish to fine-tune the parameters of a ruling line removal algorithm for a specific document image collection, as well as to researchers that develop a new ruling line removal algorithm and wish to estimate appropriate values of its parameters, evaluate it and compare it to other similar algorithms. The ground truth images may be synthetic (representing some specific type of ruling lines or just a general case) or real (e.g., created by a user who wants to adapt the algorithm to a specific document collection).

In our future plans we hope to deal with cases like that of figures 1&8, where different sets of parameters, here for horizontal and vertical ruling lines, should be combined in order to succeed the optimal result.

REFERENCES

[1] R. Cao and C. L. Tan, "Separation of overlapping text from graphics", Proc. 6th Intl. Conf., Document Analysis and Recognition, 44–48, 2001.

[2] X. Ye, M. Cheriet, and C.Y. Suen, "A generic method of cleaning and enhancing handwritten data from business forms", International Journal on Document Analysis and Recognition, 4(2): 84-96, 2001.

[3] K. R. Arvind, J. Kumar, A. G. Ramakrishnan, "Line removal and Restoration of Handwritten strokes", Intl. Conf. on Computational Intelligence and Multimedia Applications, 3: 208–214, 2007.

[4] F. Cesarini, M. Gori, and S. Marinai, "INFORMys: A Flexible Invoice-Like Form-Reader System", IEEE Trans. Pattern Analysis and Machine Intelligence, 20(7): 730-745, 1998.

[5] Y.Y. Tang, C.Y. Suen, C.D. Yan, and M. Cheriet, "Financial Document Processing Based on Staff Line and Description Language", IEEE Trans. Systems, Man and Cybernetics, 25(5): 738-753, 1995.

[6] K. Tombre, S. Tabbone, L. Plissier, and B. Lamiroy, "Text/graphics separation revisited", Workshop on Document Analysis Systems, 200–211, 2002.

[7] D. S. Doermann and A. Rosenfeld, "Recovery of Temporal Information from Static Images of Handwriting", International Journal of Computer Vision, 52(1-2): 143–164, 1994.

[8] Y. Zheng, C. Liu, X. Ding, and S. Pan, "Form frame line detection with directional single-connected chain", Proc. 6th Intl Conf. on Document Analysis and Recognition, 699–703, 2001.

[9] K. Arvind, J. Kumar, and A. Ramakrishnan, "Line removal and restoration of handwritten strokes", Intl Conf. on Computational Intelligence and Multimedia Applications, 3: 208–214, 2007.

[10] D. Dori and W. Liu. "Sparse pixel vectorization: An algorithm and its performance evaluation", IEEE Trans. Pattern Anal. Mach. Intell., 21(3): 202–215, 1999.

[11] W. Abd-Almageed, J. Kumar, and D. Doermann, "Page Rule-Line Removal Using Linear Subspaces in Monochromatic Handwritten Arabic Documents", 10th International Conference on Document Analysis and Recognition, 768-772, 2009.

[12] H. Cao, R. Prasad, and P. Natarajan "A stroke regeneration method for cleaning rule-lines in handwritten document images", MOCR, pp.1-10, 2009.

[13] Y. Zheng, C. Liu, X. Ding, and S. Pan, "Form frame line detection with directional single-connected chain", Proceedings of 6th International Conference on Document Analysis and Recognition, 699–703, 2001.

[14] G. P. G. Dimauro, S. Impedovo and A. Salzo, "Removing underlines from handwritten text: An experimental investigation. In A. C. Downton and S. Impedovo, editors", Progress in Handwriting Recognition, World Scientific Publishing, 1997.

[15] J. Said, M. Cheriet, and C. Suen, "Dynamical morphological processing: a fast method for base line extraction", In ICDAR, 8–12, 1996.

[16] X. Ye, M. Cheriet, and C. Y. Suen, "A generic method of cleaning and enhancing handwritten data from business forms", Int. J. on Document Analysis and Recognition, 4:84-96, 2001.

[17] Z. Shi, and V. Govindaraju, "Line separation for complex document images using fuzzy runlength", Proceedings of First Intl. Workshop on Document Image Analysis for Libraries, 306-312, 2004.

[18] Z. Shi, S. Setlur and V. Govindaraju, "Removing Rule-lines From Binary Handwritten Arabic Document Images Using Directional Local Profile", Intl. Conf. Pattern Recognition, 1916–1919, 2010.

[19] E. Giuliano, O. Paitra, and L. Stringer, "Electronic character reading system",U.S. Patent No. 4,047,15 Sep. 1977.

[20] E.R. Davies, "Machine Vision: Theory, Algorithms, Practicalities", 3rd Edition, ELSEVIER, 164 – 167, 2005.

[21] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, "Equation of state calculations by fast computing machines", *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[22] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi. "Optimization by simulated annealing", *Science*, 220(4598):671–680, 1983.

[23] V. Černý, "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm", *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985. [58] W.H.

[24] Press, S.A. Teukolsky, W.T. Vettering, and B.P. Flannery, "Numerical Recipes in C++. Example Book. The Art of Scientific Computing", *Cambridge University Press, 2nd edition, chapter 10*, 2002.

[25] E. Kavallieratou, D. Lopresti and J. Chen, "Ruling Line Detection and Removal", *Document Recognition and Retrieval XVIII, IS&T/SPIE Electronic Imaging*, 2011.