NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

FACULTY OF EXACT SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

PAPER

A Morphological Approach to Text Extraction from Images with Pattern-Based Background

Orestis Melkonian

Supervisor: **Basilis Gatos,** Researcher at Centre for Scientific Research Demokritos

ATHENS

MAY 2015

PAPER

A Morphological Approach to Text Extraction from Images with Pattern-Based Background

Orestis Melkonian
A.M.: 1115201000128

SUPERVISOR: **Basilis Gatos,** Researcher at Centre for Scientific Research Demokritos

ABSTRACT

The vast majority of OCR systems operate on clean images of black text on a white background. Unfortunately, this does not hold for real-world applications, as most documents are transferred in a more complex manner. For example, forms completed by hand will reduce the efficiency of an OCR system by carrying the information of the form pattern, when the system only cares about the letters that have been written.

In this paper, I provide documentation for a software program I developed, which can be used for extracting text written or printed in a sheet of paper containing a specific pattern, repeated at fixed intervals both horizontally and vertically.

In contrast with prior work on similarly themed topics, which mainly use standard singal processing techniques, I utilized mathematical morphology, which is a perfect fit for shape-based operations and allow for a very simple, efficient and clean implementation.

Experimental evaluation was done by calculating the F-Measure of given images of variable font size and randomly generated patterns on the background.

SUBJECT AREA: Document Analysis

KEYWORDS: text extraction, mathematical morphology, image processing, pattern removal,

ACKNOWLEDGEMENTS

I would like to thank Mr. Basilis Gatos for the opportunity to work with such an interesting topic.

CONTENTS

PROLOGUE

The current document was written in Athens, in 2015, within the context of a project for the undergraduate course D̈ocument Analysis - Optical Character Recognition.¨

1. INTRODUCTION

This paper documents the development of a software program, which is a direct implementation of the algorithm proposed by Liang in 1994 [1]t. The necessity of such a tool is necessary as a preprocessing procedure on a given document before a recognition algorithm is applied.

There has been extensive research on the extraction of text intersecting with geometric patterns [3, 5, 6, 7]. Thresholding [7, 8, 9] is a dominant approach as it already has been developed for image binarization. In addittion, eliminating noise background [5] would increase the efficiency of the OCR system considerably. Billawala [4] described a technique called the *image continuation algorithm* to remove the scratches and blemishes in binarized images. Ozawa [3] proposed a method to remove the geometric pattern background in Japanese newspaper headline.

The program developed focuses on binarized documents containing text with a regular periodic overlapping background. This is also the reason why mathematical morphology, because of its ability to grasp the geometry and structure of images, is the best choise for this special operation.

2. DEFINITIONS

In this section, the principles of mathematical morphpology as well as the basic morphological operations will be defined.

2.1 Principles of Mathematical Morphology

Mathematical morphology is a set-theoretic approach to image processing, fathered by Jean Serra [2], which analyses and processes geometrical structures based on set theory, lattice theory, topology and random functions. It considers images to be sets in underlying space and manipulates them using set-based operation such as union and intersection. Application is made possible by defining a structuring element and using a morphological operator on a given image with this element.

Image A NxN matrix of zeros and ones.
Structuring element A NxN matrix of zeros, ones and don't cares (0, 1, *) with an origin point.
$(X)_y$ The translation of binary image X by vector y.

2.2 Morphological Operators

2.2.1 Union

Let I_1 and I_2 be two images.

$$I_1 \bigcup I_2 = \{x | I_1(x) = 1 \vee I_2(x) = 1\} \tag{1}$$

2.2.2 Intersection

Let I_1 and I_2 be two images.

$$I_1 \bigcap I_2 = \{x | I_1(x) = 1 \wedge I_2(x) = 1\} \tag{2}$$

2.2.3 Erosion

Let I be an image and S be a structuring element.

$$I \ominus S = \bigcap_{b \in B} (X)_{-b} \tag{3}$$

Erosion is a shrinking of the original image.

2.2.4 Dilation

Let I be an image and S be a structuring element.

$$I \oplus S = \bigcap_{b \in B} (X)_b \tag{4}$$

Dilation is an expansion of the original image.

2.2.5 Opening

Let I be an image and S be a structuring element.

$$I \circ S = (X \ominus B) \oplus B \tag{5}$$

Opening is anti-extensive and rounds off things, so sharp edges disappear.

2.2.6 Closing

Let I be an image and S be a structuring element.

$$I \bullet S = (X \oplus B) \ominus B \tag{6}$$

Closing is an expansive and removes small holes in shapes.

3. METHODOLOGY

3.1 Stage 1: Calculate periodicity of the background symbols

At this stage, we want to find the exact horizontal and vertical distance between the symbols of the background.

PDH Periodic distance in the horizontal direction.
PDV Periodic distance in the vertical direction.

The above satisfy the following inequalities:

$$CL((X \ominus T1) \ominus B_{PDH}) > CL((X \ominus T1) \ominus B_i), i = 1, 2, .., M, i \neq PDH$$
$$CL((X \ominus T2) \ominus B_{PDV}) > CL((X \ominus T2) \ominus B_j), j = 1, 2, .., M, j \neq PDV$$

where X is the given image, $CL(E)$ is a pixel-counting function which calculates the total number of ones in the image and B_i, B_j are two point pair structing elements of length i, j repsectively.
$T1$ and $T2$ erode the image set X to extract left and top edges.

Computationally, the calculation of PDH and PDV is equivalent to finding the maximum number of pixels when the image is eroded with the B_{PDH}, B_{PDV} respectively.

3.2 Stage 2: Recognize Pattern

The goal of this stage is to extract the background components, so as to remove them later.
This is achieved by designing the structuring elements $S1$, $S2$, $S3$ and $S4$ to realize the appropriate morphological erosion operations. These operations are used to remove the text from the background image.
$S1$ wil erode pixels that do not have a horizontal right matching pixel, $S2$ will erode pixels that do not have a horizontal left matching pixel, etc...
We repeatedly apply the below erosion till a minimum value of eroded pixels is reached:

$$((((X \ominus S1) \ominus S3) \ominus S2) \ominus S4) \tag{7}$$

3.3 Stage 3: Remove Pattern

So far, we managed to extract the image of just the background symbols, without any text. In order to remove the pattern, we simply XOR the original image X with the background image Y, which is the result of Stage 2.

$$Z = XOR(X, Y) \tag{8}$$

The result is not nearly satisfying as any intersection of the text with the background symbols will result in empty gaps inside the characters.

3.4 Stage 4: Optimization

To overcome the above drawback and improve the quality of the result, a conditional dilation followed by a closing operation are performed as the below equation shows.

$$W = ((((Z \oplus B1) \cap Y) \cup Z) \bullet B2) \tag{9}$$

where B1 and B2 are properly selected structuring elements.

4. IMPLEMENTATION

4.1 Environment

The program is written in the C++ programming language, under the *Borland C++ Builder* IDE. This particular IDE provides an easy API for desktop graphic applications, which really helped to minimize technical burdens and allow for essential research.

To enable image processing, the *ImagXpress v7, Pegasus Imaging* plugin must be installed in the IDE.

The text images where produced in *Microsoft Word 2007*.

4.2 User Interface

4.2.1 Menu Commands

The current structuring element can be set in the *Settings* window, by pressing the *Settings* button in the lower right part of the screen.

A description of the commands in the menu bar follows:

File→Open Image... Opens an image that exists on the file system.

File→Save Image Saves the image inside the current directory.

File→Binary Convert Converts current image to binary.

Steps→Find PDH/PDV Calculates the periodocity of the background symbols.

Steps→Find pattern Removes the text from the image, resulting in the background image.

Steps→Remove pattern Removes the background from the initial image.

Steps→Optimize Increases the quality of the result of the last step.

Morphology→Dilation Performs *dilation* of the image with the current structuring element.

Morphology→Erosion Performs *erosion* of the image with the current structuring element.

Morphology→Opening Performs *opening* of the image with the current structuring element.

Morphology→Closing Performs *closing* of the image with the current structuring element.

Generate→Add random pattern Draws a random pattern on the given image.

Generate→Add lines Draws a standard line-based pattern on the given image.

Evaluation→Get score Calculates the *F-measure*[10] at the current step.

4.2.2 Usage Scenario

An indicative usage scenario follows:

1. File→Open Image...
2. Generate→Add random pattern/Add lines
3. Steps→Find PDH/PDV
4. Steps→Find pattern
5. Steps→Remove pattern
6. Steps→Optimize
7. File→Save Image

5. EXPERIMENTAL EVALUATION

5.1 Preparation

As input, we will give three text images of small/medium/large font size respectively. Each will be tested with four pattern backgrounds (dot-based, line-based, cross-based, random). The quantification stage will be done using the F-Measure of each image, which is a function over the Recall and Precision values [10]. As we an clearly see, quality decreases as we move to higher pattern size - font size ratio values. Another reason for image degradation is high density of the pattern, that is smaller values for the PDH and PDV. Results are displayed below.

5.2 Results

5.2.1 Large Font

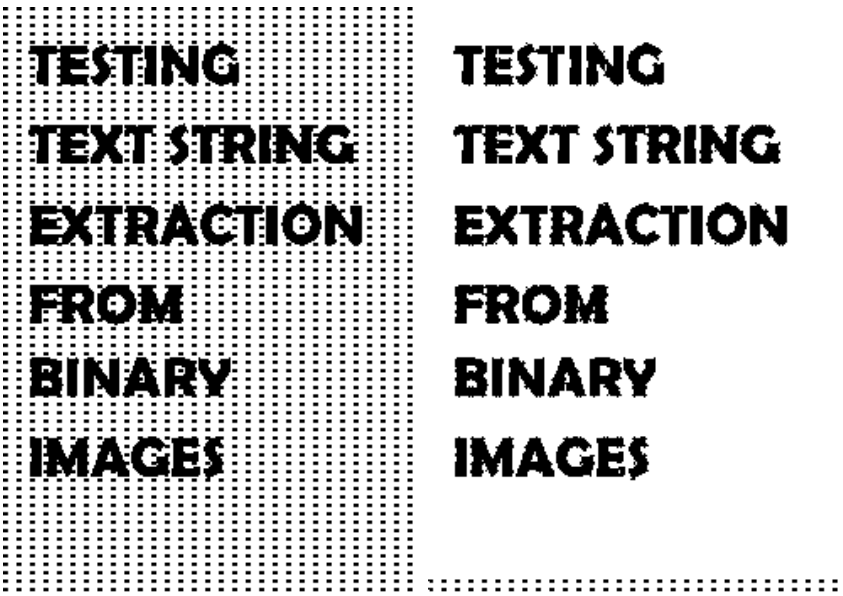

Fig. 1: Dots - 95,95%

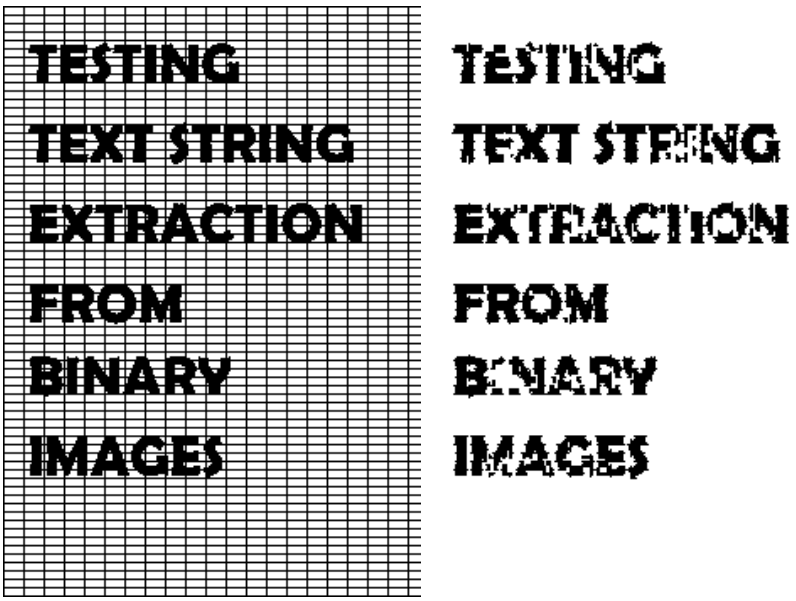

Fig. 2: Lines - 90,02%

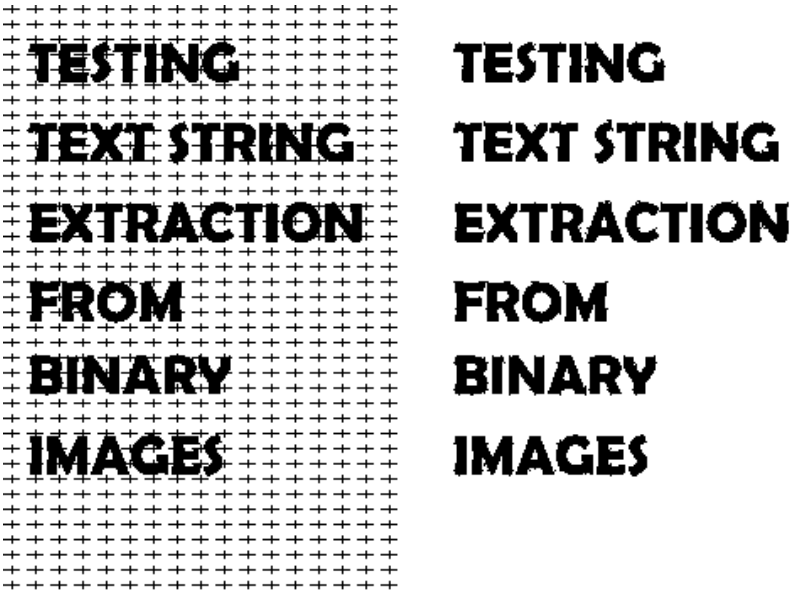

Fig. 3: Crosses - 98,86%

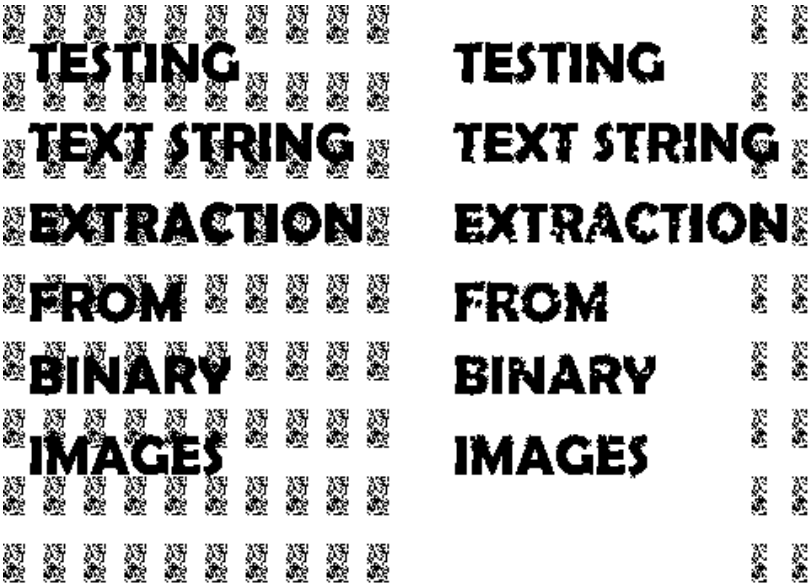

Fig. 4: Random - 90,02%

5.2.2 Medium Font

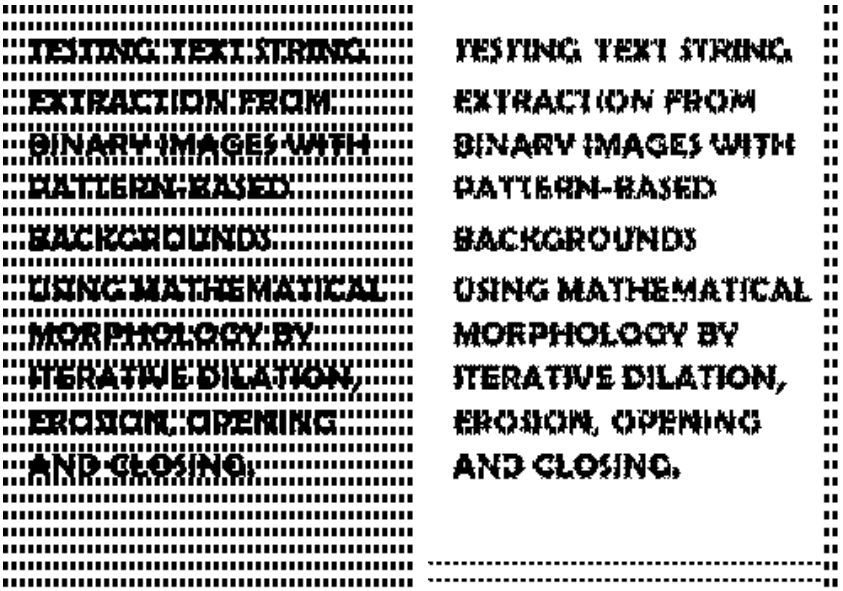

Fig. 5: Dots - 84,79%

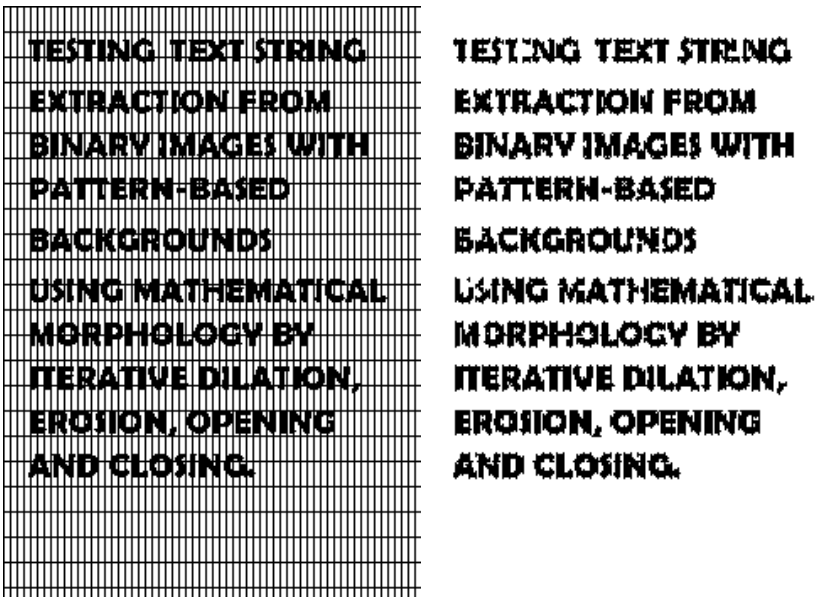

Fig. 6: Lines - 90,17%

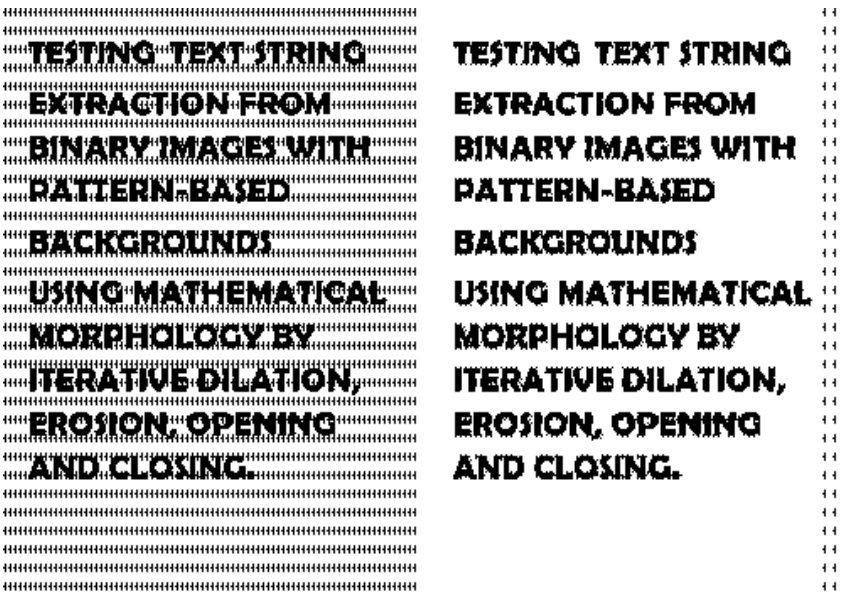

Fig. 7: Crosses - 93,39%

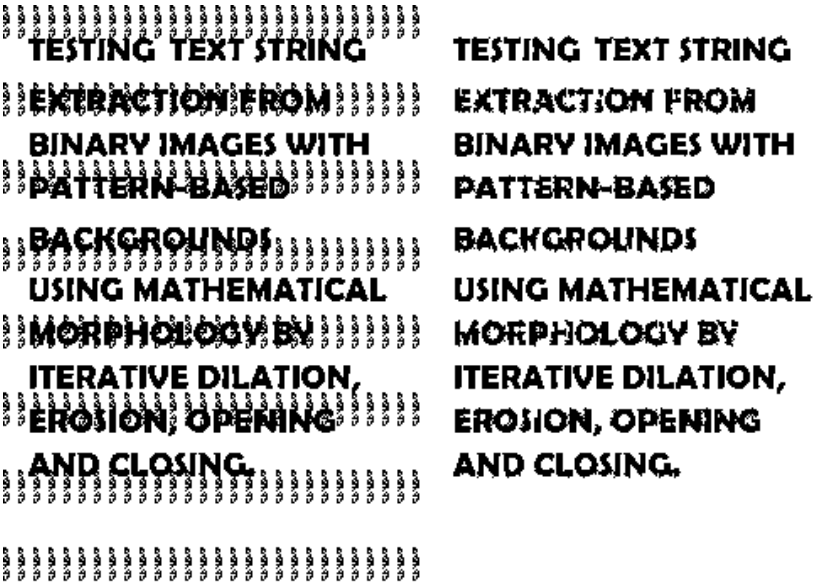

Fig. 8: Random - 96,81%

5.2.3 Small Font

Fig. 9: Dots - 96,63%

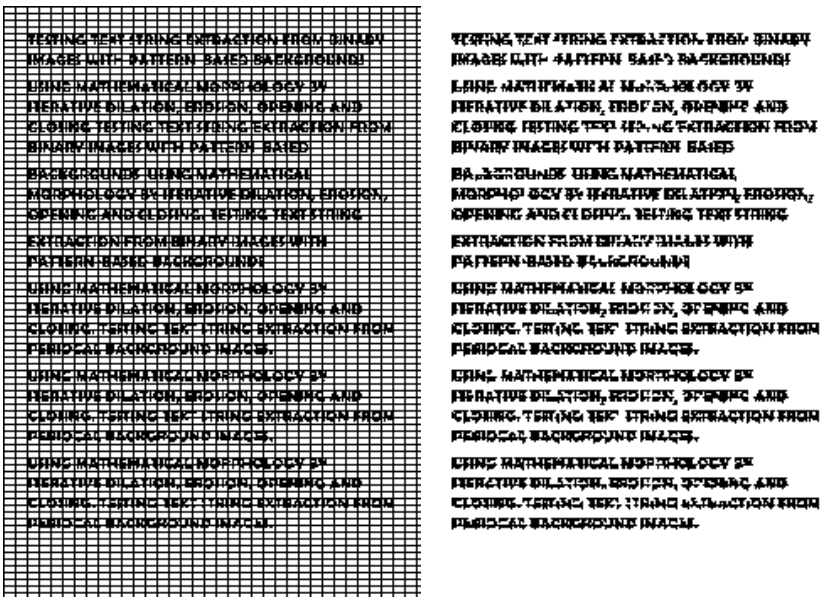

Fig. 10: Lines - 79,68%

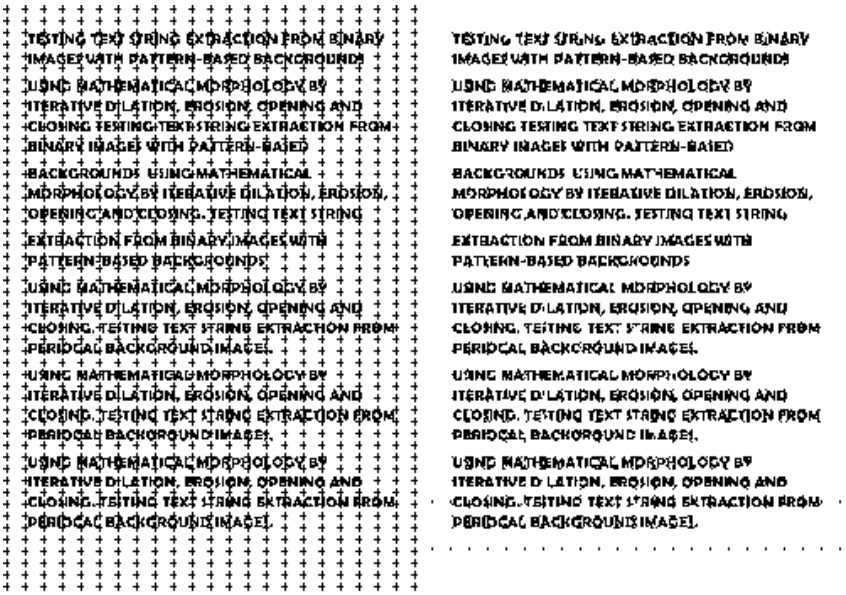

Fig. 11: Crosses - 93,47%

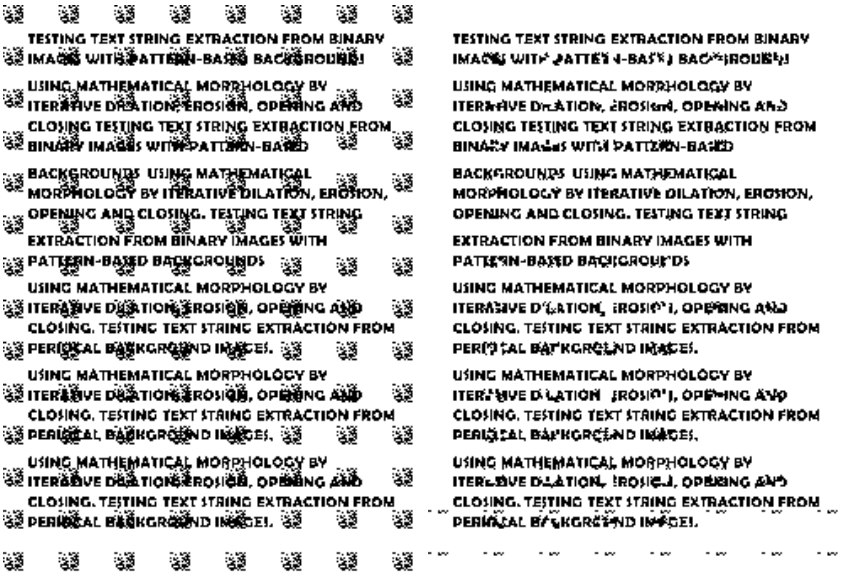

Fig. 12: Random - 94,86%

6. SIMILAR WORK

There has been remarkable work on similar problems, some utilizing mathematical morphology and others that do not.

My research was based on two surveys on generic text extraction [11, 12].

Hasan and Karam [13] have written by far the most citated paper on text extraction using mathematical morphology which is not restricted to special occurences of background, such as noise, skew and non-text artifacts. Their method is heavily based on previously developed robust edge extraction techniques using mathematical morphology [14], in combination with a new region segmentation algorithm.

An interesting case of a special problem is that of extracting text from business forms [15]. This paper proposed a generic system including only cleaning and enchancing phases, so as to output a more proper input to a following OCR system. A remarkable unified morphological scheme is designed to remove the form frames and also restore broken handwriting.The real problem is when handwriting touches or crosses pre-printed text. To overcome this, morphological operations based on statistical features are used.

A more advanced use of morphology can be seen in Gu's [16] proposal for character extraction from color images. This paper utilized a more complex morphological operation, called *Top-Hats Transformation*, first proposed by Meyer [17], along with iterative morphological operations.

Coupling morphology with text features, such as edges and clusters [19] is also a famous technique, as previous work can be easily utilized. The strong point of this paper is that it has great results even in real-world scenes, which are characterized by enormous complexity.

A cutting-edge application was developed by Ma, Lin and Zhang [18, 22] for the Android platform, which recognises text captured by a mobile phone camera, translates the text and display the translated result back onto the screen of the mobile phone.

An out-of-the-box approach is that of Singh and Khare [20], where text region extraction is achieved by combining mathematical morphology with genetic algorithms, which are generative procedures that simulate natural selection by using standard operations from the Genetic field, such as cross-over, mutation and fitness functions.

Moving to a broader topic, mathematical morphology has also been proven helpful to the problem of inpainting. Inpainting is the process of reconstructing lost or deteriorated parts of images and videos. Specifically, what is interesting in the context of this paper, is how inpainting can be used to remove text from images and restore the original image. Vaghela and Patel [22] use morphological operations for formatting the text regions and then text feature filtering for to extract the text string. Modha and Dave [22] use morphology for edge detection and then feature filtering for the remaining stages.

7. CONCLUSION

In this paper, i document the direct implementation of Liang's paper [1] in C++ under the Borland C Builder IDE. The principles of mathematical morphology are given, as this method relies almost completely on morphological operators. Additionally, a detailed description of the procedure followed is given, along with standard usage scenarios and explanation of the User Interface. To conclude, international work on similar topics is discussed and the importance of the mathematical morphology field is again emphasized.

ABBREVIATIONS

OCR	Optical Character Recognition
PDH	Periodic Distance Horizontally
PDV	Periodic Distance Vertically
IDE	Integrated Development Environment
API	Application Programming Interface

REFERENCES

[1] Liang, Su, Ahmadi, M., Shridhar M.: *A morphological approach to text string extraction from regular periodic overlapping text/background images*. Proc. of IEEE Int. Conf. on Image Processing, (ICIP-94) (1994) 144-148

[2] J. Serra: *Image analysis and mathematical morphology*. Academic, London, 1982

[3] H. Ozawa and T. Nakagawa: *A character image enchanement method from characters with various background images*. Proceedings, Second International Conference on Document Analysis and Recognition Tsukuba, Japan, Oct. 1993, pp. 58-61.

[4] N. Bilawala, P. Hart and M. Peairs: *Image Continuation*, Proceeding, First International Conference on Document Analysis and Recognitionm Saint-Molo, France, Sept.30-Oct.2, 1991, Vol.2, pp. 993-999.

[5] R. M. Lougheed: *An overview of grayscale morphological filters* in 23th Asilomar Conference on Signals, Systems and Computers, Oct.30-Nov.1, Pacific Grove, California, 1989, Vol. 1, pp. 152-156.

[6] H. Yamada, K. Yamamoto, T. Saito, K. Hosokawaa and H. Yanagisawa: *Laser-marked alphanumeric character recognition by multi-angled matching method* in 11th IARP International Conference on Pattern Recognition, The Hague, The Netherlands, Aug.30-Sept.3, 1992, pp. 326-329.

[7] J. M. White and G. D. Rohrer: *Image thresholding for optical character recognition and other applications requiring character image extraction*, IBM J. Res. Dev. 27(4), 1983, 400-410.

[8] R. C. Gonzalez and R. E. Woods: *Digital Image Processing*, Addison-Wesley, 1992.

[9] J. Kittler and J. Illingworth: *Minumum error thresholding*, Pattern Recognition, 29(I), 1986, 41-47.

[10] Powers and David M W: *Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation*. 2011, Journal of Machine Learning Technologies 2 (1): 37–63.

[11] Keechul Jung, Kwang In Kim, Anil K. Jain: *Text Information Extraction in Images and Video: A Survey*.

[12] A. J. Jadhav, Vaibhav Kolhe and Sagar Peshwe: *Text Extraction from Images: A Survey*, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 3, March 2013. [Online]. Available: http://www.ijarcsse.com.

[13] M. Y. Hasan and J. Karam: *Morphological Text Extraction from Images*, IEEE Transactions on Image Processing, Vol. 9, No. 11, November 2000.

[14] K. K. Chin and J. Saniie: *Morphological processing for feature extraction*, Proc. SPIE, vol. 2030, pp. 288–302, 1993.

[15] X. Ye, M. Cheriet and Y. Suen: *A generic method of cleaning and enchancing handwritten data from business forms*, IJDAR (2001) 4: 84–96.

[16] L. Gu, T. Kaneko, N. Tanaka and R. M. Haralick: *A New Morphological Segmentation Method Applied for Character Extraction*, Proceeding ISMM ’98 Proceedings of the fourth international symposium on Mathematical morphology and its applications to image and signal processing Pages 367-374.

[17] F. Meyer: *Constrast Feature Extraction*, Quantitative Analysis of Microstructures in Material Sciences, Biology and Medicine. J-L Chermant, ed. Special Issue of Practical Metallography, Riederer Verlag, Stuttgart, Germany, 1978.

[18] D. Ma, Q. Lin and T. Zhang: *Mobile Camera Based Text Detection and Translation*, Stanford University.

[19] W. Josephin and Dr. R. K. Selvakumar: *An Edge Based Text Segmentation From Complex Images*, International Journal of Engineering Research and Technology (IJERT), Vol. 2 Issue 7, July 2013.

[20] D. P. Singh and A. Khare: *Text Region Extraction: A Morphological Based Image Analysis Using Genetic Algorithm*, I.J. Image, Graphics and Signal Processing, 2015, 2, 39-47.

[21] U. Modha and P. Dave: *Image Inpainting-Automatic Detection and Removal of Text From Images*, Uday Modha, Preeti Dave / International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 2,Mar-Apr 2012, pp.930-932.

[22] K. Vaghela and N. Patel: *Automatic Text Detection using Morphological Operations and Inpainting*, International Journal of Innovative Research in Science, Engineering and Technology, Vol. 2, Issue 5, May 2013.