Identification of Text on Colored Book and Journal Covers

Karin Sobottka　Horst Bunke　Heino Kronenberg

Institute of Informatics and Applied Mathematics
University of Bern, Neubrückstrasse 10
CH-3012 Bern, Switzerland
Phone: +41-31-6314987　Fax: +41-31-6313965
E-Mail: {sobottka,bunke,kronenbe}@iam.unibe.ch

Abstract

In this paper an approach to automatic text location and identification on colored book and journal covers is proposed. To reduce the amount of small variations in color, a clustering algorithm is applied in a preprocessing step. Two methods have been developed for extracting text hypotheses. One is based on a top-down analysis using successive splitting of image regions. The other is a bottom-up region growing algorithm. The results of both methods are combined to robustly distinguish between text and non-text elements. Text elements are binarized using automatically extracted information about text color. The binarized text regions can be used as input for a conventional OCR module. Results are shown for several book and journal covers of different complexity. The proposed method is not restricted to book and journal cover pages, but can be applied to the extraction of text from other types of color images as well.

KEYWORDS: automatic text location, information retrieval, color image processing

1 Introduction

Most optical character recognition (OCR) techniques are restricted to greylevel or binary images of text. But there are many applications where text is embedded in color images. To apply an OCR system to such an image, a preprocessing step is needed that automatically identifies text regions and maps them into the monochromatic color space. However, automatic identification of text is a difficult task since not only color, character font and size, but also alignment and orientation of text elements may change on the same page. Colors that appear uniform to a human observer show many small variations after digitalization. Furthermore the presence of texture or objects in the background increase the difficulty of text identification.

In this paper we consider the problem of automatic text identification on book and journal cover pages. As input data we consider images scanned with a resolution of 200 dpi and a color depth of 24 bit. The images typically have a size of about 1300 x 1800 pixels and include thousands of colors. To reduce the number of colors, a graph-theoretical clustering algorithm is applied in a preprocessing step. Each pixel of the preprocessed image is represented by the mean color of the cluster it was assigned to. Two methods are described for the establishment of hypotheses about text elements. The top-down analysis is based on splitting image regions alternately in horizontal and vertical direction. The regions resulting from this process are always of rectangular shape. Since a rectangular shaped region containing text includes also pixels from the background and thus has at least two different colors, homogeneous regions are rejected as non-text elements during segmentation. On the contrary, the bottom-up analysis intends to find homogeneous regions of arbitrary shape. A region growing technique is applied to merge pixels belonging to the same cluster. Assuming that text is aligned horizontally, both segmentation schemes try independently of each other to group regions of interest into lines. The resulting hypotheses for text elements are verified by combining the results of both methods. Elements identified as text are binarized using the text color, which is automatically extracted.

In section 2 related work on the automatic location and identification of text elements in paper documents and images is described. The graph-theoretical clustering algorithm for preprocessing is explained in section 3. Outlines of the top-down and bottom-up analysis are given in sections 4 and 5, respectively. The composition of regions into lines and the selection of text candidates is described in section 6. In section 7 the identification and binarization of text elements is outlined. Results are shown in section 8. The paper is concluded with a summary.

2 Related Work

The literature on text location in document images is not too abundant. In the following a brief overview is given.

Several methods for text location are discussed in [3]. In addition to multivalued image decomposition and foreground image generation and selection, color space reduction and text location using statistical features is proposed. Applications are presented which include text location in advertisements, Web images, general color images, and video frames.

In [8] an approach for locating and extracting text from Web images is described. First color clustering using the Euclidean minimum-spanning-tree is performed. Secondly, connected component analysis is applied. Text-like connected components are identified in each color class based on their shape. Finally text-like components are aligned into lines.

The problem of video indexing is addressed in [6]. First a sub-pixel interpolation technique is applied to obtain a higher resolution for text regions. Then four filters are employed to extract horizontal, vertical, left diagonal and right diagonal line elements. Since pixels for characters are expected to have high filter responses, a thresholding step is performed next. It is assumed that captions have high intensity values. Hence, peaks in the vertical projection profile define the boundaries between characters. Oversegmentation is reduced by a postprocessing step.

Two methods for automatically locating text in complex color images are described in [7]. The first method segments the image into connected components with uniform color, and uses several heuristics such as size, proximity and alignment to select character-like components. The second method computes the spatial variation in the grey-scale image, and locates text in regions with high variance. Afterwards the results of both approaches are combined. The methods are used to locate text in compact disc and book cover images, as well as in images of traffic scenes captured by a video camera.

A method for detecting Japanese characters in grey-scale images of scenes is proposed in [4]. In a first step subregions with high spatial frequency and large variance in grey-level values are extracted as character candidates. Then, characters are identified by using several heuristics such as size, shape, bimodality of the intensity histogram, alignment, and proximity. The system is employed for mobile robot navigation by character information.

Further approaches are concerned with location of address blocks on letters [2] and extraction of filled-in data from color forms [1].

3 Preprocessing

A color picture that appears uniform to a human observer will show many small variations in color after digitalization. These variations are not very useful and should be eliminated in a preprocessing step. In our approach to text extraction from colored book and journal covers, we use an unsupervised clustering algorithm for preprocessing. It is intended to find clusters of similar colors. Based on the clusters, a preprocessed image is computed in which pixels are labelled with the same color value if their original colors belong to the same cluster. Investigations were done on both the red-green-blue (RGB) and hue-saturation-value (HSV) color space. Although the HSV color space is more compatible with human color perception, we obtained better results using the RGB color space.

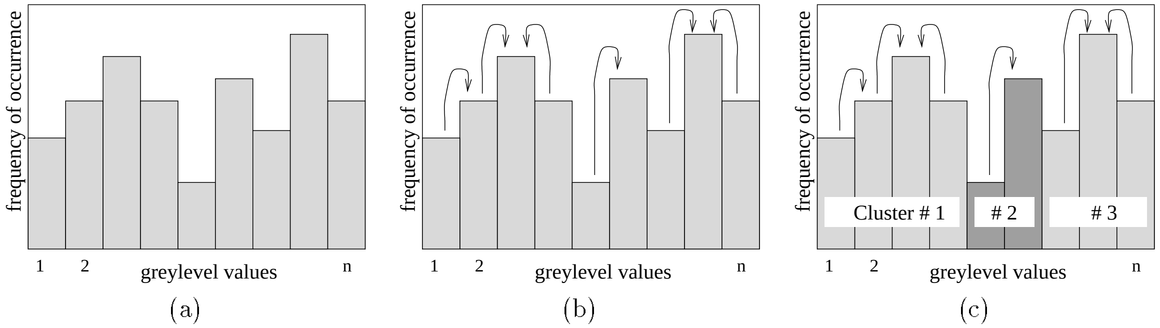
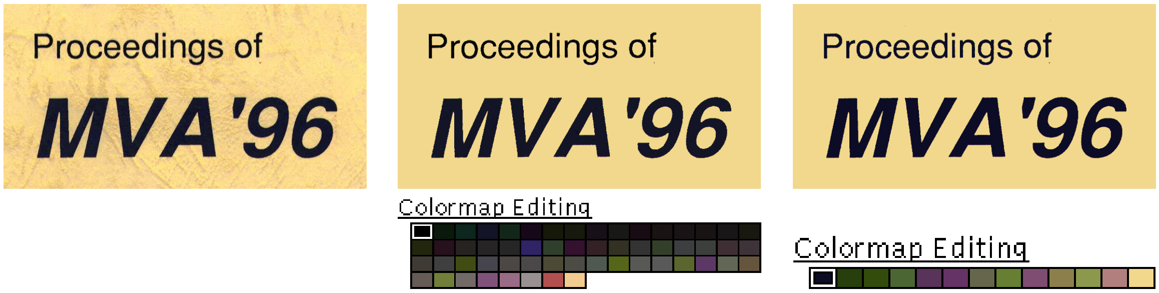


Figure 1: Example for graph-theoretical clustering applied to greylevel image: (a) histogram, (b) chains of pointers and (c) clusters

The clustering algorithm used in our approach is based on the method described in [5]. The fundamental idea can be described as follows: In a first step statistical information is evaluated by building a histogram (Fig. 1a). Secondly, for each cell in the histogram a pointer to its largest neighbor cell is stored (Fig. 1b). After pointers are set for the whole histogram, the histogram contains chains of cells pointing to a local maximum. The set of cells belonging to such a chain build a cluster (Fig. 1c). The example shown in Fig. 1 relates to the case of a greylevel image. For applying the graph-theoretical clustering to a color image a 3D histogram has to be computed and for each cell 27 neighbors evaluated. Experiments showed that the number of resulting clusters depends on the color space and the quantization of colors. Due to the hexcone shape of the HSV color space, the number of clusters is generally larger than for the RGB color space. The results on a part of an example image is shown in Fig. 2. In Fig. 2a the original image part containing more than 150 000 colors is shown. The result of graph-theoretical clustering done in the HSV and RGB color space is shown in Fig. 2b,c, respectively. A color quantization of 15x15x15 leads to 56 clusters in case of the HSV color space. Using the RGB color space the number of clusters is significantly reduced (13 clusters). The textured background is represented by one cluster for both color spaces.

4 Top-Down Analysis

To extract text candidates, the image is alternately split in horizontal and vertical direction. Regions obtained under this top-down procedure are always of rectangular shape, and regions containing text



> 150 000 colors 56 colors (HSV) 13 colors (RGB)

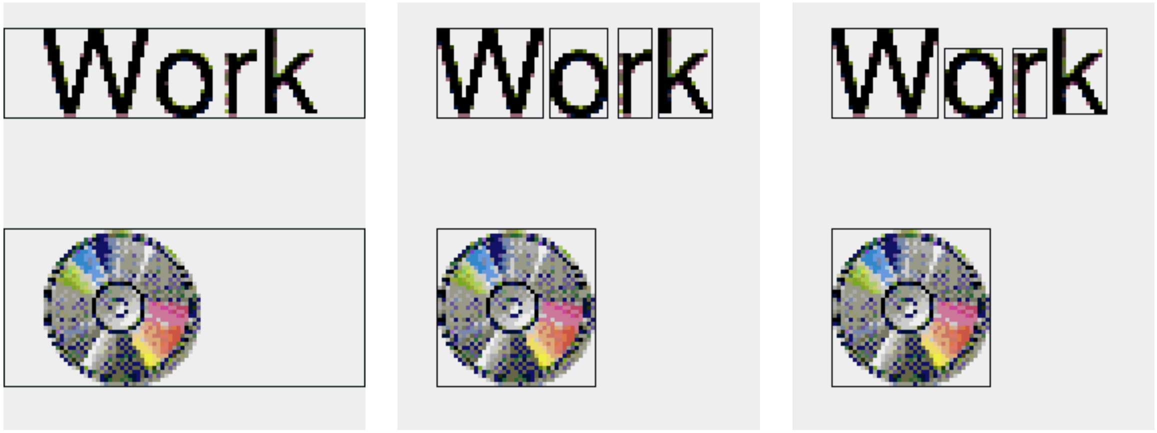
Figure 2: Results of preprocessing: (a) part of original image, (b) clustered in HSV color space and (c) clustered in RGB color space

include at least two colors. We use this knowledge to reject homogeneous regions as non-text elements during segmentation.

4.1 Procedure

Beginning with the whole image as start region, each region is processed in an iterative manner. In one iteration step a region is split horizontally or vertically. The direction of splitting alternates for each iteration. Several partitions of an inhomogeneous region into homogeneous and inhomogeneous subregions are possible in one iteration step. Inhomogeneous subregions are considered as possibly containing text elements and thus splitting continues. Homogeneous subregions are rejected as non-text regions. For inhomogeneous regions splitting terminates if at least two cluster labels arise for all rows and columns of the region. During splitting a region, information about possible text and background colors is acquired. This information is used to avoid oversegmentation of characters.

An example for successively splitting part of a book cover image is shown in Fig. 3. The result after the first, second and third iteration step is given. As can be seen in Fig. 3a the first split is in horizontal direction. Splitting continues for the two inhomogeneous regions which contain text and a graphic element. Fig. 3b shows the result for vertically splitting (second iteration). The word is split into characters and the graphic element is cut at its vertical borders. In the third iteration splitting terminates for the region containing the graphic element since all lines and columns consists of more than one label. The four regions containing characters are further partitioned horizontally (Fig. 3c).

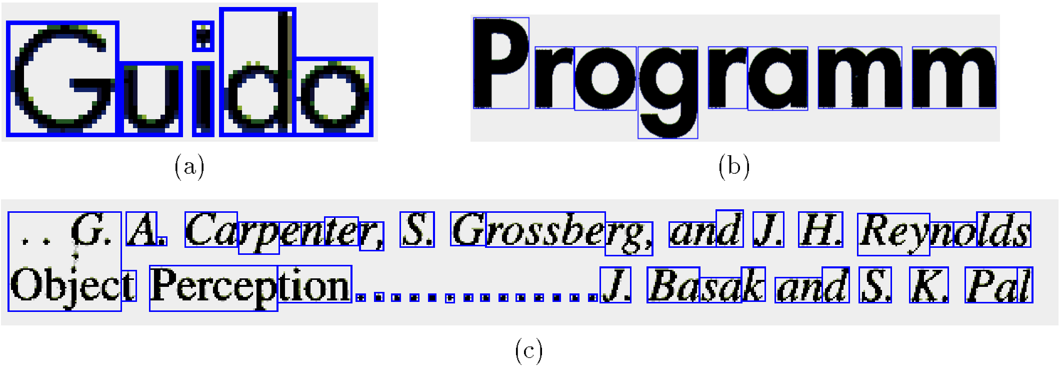


(a) first iteration (b) second iteration (c) third iteration

Figure 3: Results of top-down analysis after (a) first, (b) second and (c) third iteration

4.2 Characteristics

The top-down analysis using successive splitting shows the following characteristics. Small sized text is segmented very accurately even if noise pixels are present at the contour of text. An example for the segmentation of text printed with a size of 11 pt is depicted in Fig. 4a. As can be seen several colors are present at the contour of the word “Guido”. Nevertheless characters are detected entirely. Moreover, using successive splitting characters are not oversegmented. Thus each region corresponding to text contains at least one character. An example is shown in Fig. 4b, where each region contains one character.


Guido
Programm
.. G. A. Carpenter, S. Grossberg, and J. H. Reynolds
Object Perception................J. Basak and S. K. Pal

(a)

(b)

(c)

Figure 4: Characteristics of top-down analysis: (a) small sized text is segmented accurately, (b) characters are not oversegmented and (c) characters may be undersegmented

In case that text is printed in italics, or in presence of noise, an undersegmentation may occur. An example is shown in Fig. 4c, where several regions contain more than one character. Our successive splitting procedure is furthermore characterized by the fact that regions containing background images or graphic elements are typically not split into subregions.
Measurements of the computing time shows that the top-down analysis is quite efficient. The mean processing time for the segmentation of a book or journal cover with a resolution of 1600 x 2400 pixels is about 1.05 sec on a SUN Ultra-Sparc 5/10 workstation.

5 Bottom-Up Analysis

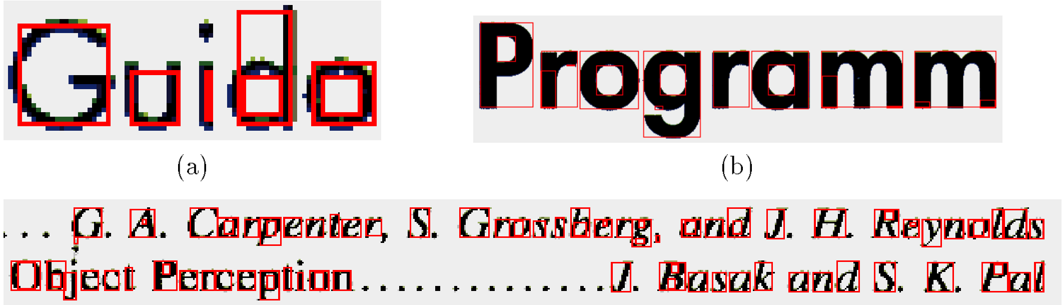
The bottom-up analysis detects homogeneous regions using a region growing method. Beginning with a start region, pixels are merged if they belong to the same cluster. Since characters of machine printed text generally do not touch each other, several regions result for a text region.

5.1 Procedure

Region growing is a well known technique in image processing. We employ it to extract homogeneous regions. As start region three horizontally or vertically adjacent pixels which belong to the same cluster and are not yet assigned to a region are selected. Beginning with the start region, pixels within a 3x3 neighborhood are iteratively merged if they belong to the same cluster. The algorithm terminates when all pixels are merged to one of the regions, or no further start region can be found. The regions resulting from this procedure are represented by their bounding boxes.

5.2 Characteristics

Using bottom-up analysis the extraction of small sized text is difficult. Due to noise at text contours, often only the inner parts of characters are detected. The text depicted in Fig. 5a was printed with a size of 11 pt. As can be seen characters are detected too small. Thus oversegmentation may arise for characters. In particular for characters that include background, at least two regions are extracted, one belonging to the text and the other to the included background. Fig. 5b shows an example. Inclusions arise for the characters p, o, g, a. On the other hand characters that are not touching each other are not merged. Thus characters are always separated even if they are printed in italics. An example is depicted in Fig. 5c.


Guido
Programm
... G. A. Carpenter, S. Grossberg, and J. H. Reynolds
Object Perception .... J. Basak and S. K. Pal

(a)

(b)

(c)

Figure 5: Characteristics of bottom-up analysis: (a) small sized text is segmented too small, (b) characters are often oversegmented and (c) not touching characters are never merged

Using bottom-up analysis background images or graphic elements are typically partitioned into many regions if they include different colors. Measurements of the computing time show that the bottom-up analysis is more time consuming than top-down analysis. The mean processing time for the segmentation of a book or journal cover with a resolution of 1600 x 2400 pixels is about 20.26 sec measured on a SUN Ultra-Sparc 5/10 workstation.

Compared to top-down analysis the characteristics of bottom-up analysis are just complementary (Fig. 4, 5). We make use of this by combining the results of both for text identification.

6 Grouping of Regions

We assume that text consists of horizontally aligned lines. In order to find subsets of regions which are aligned horizontally a grouping step is applied. The regions of interest determined by top-down and bottom-up analysis are considered separately. Thus several text line candidates result for top-down as well as bottom-up analysis. In both cases the same grouping technique is applied.

In our approch text lines are assumed to consists of more than three regions having a small horizontal distance and a large vertical overlap to each other. An example for a text line composed of regions $R_1, \ldots, R_{11}$ is shown in Fig. 6.

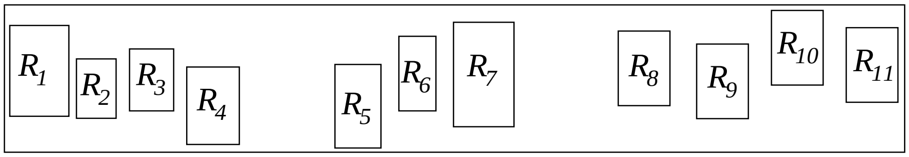


Figure 6: Example for a text line composed of regions R_i

For each pair of regions R_i and R_j it is tested whether their horizontal distance d is smaller than a predefined threshold Θ (Fig. 7a). For the definition of Θ we have taken into consideration that the horizontal distance between words is larger than between characters. Therefore we defined Θ as follows:

$$\Theta = 5 \cdot max(w(R_i), h(R_i)) \tag{1}$$

where h and w denote the height and width of a region, respectively. Furthermore it is required that the difference between the upper and lower borders of R_i and R_j are in a certain tolerance interval, which is defined dependent on the height h of a region (Fig. 7b,c). The upper or lower border of a region must vary not more than $\frac{2}{3}h$.

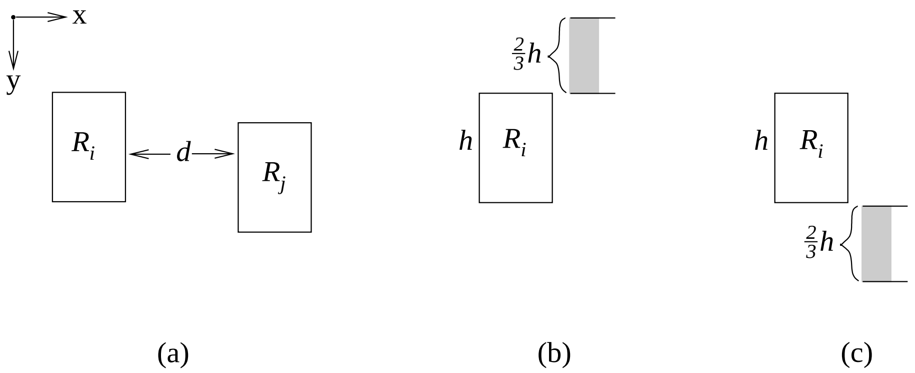


Figure 7: Features for grouping of regions

The detected regions-of-interest are represented by their bounding boxes. Results for an example image are shown in Fig. 8. In this example the book cover contains several text elements, logos and a complex image. The result of composition of regions determined by top-down analysis is shown in Fig. 8a. As can be seen text lines are well grouped. In one case, part of a logo is erroneously grouped with text. The image is represented as one region due to the characteristics of top-down analysis. Using the regions of bottom-up analysis as input, different results are obtained (Fig. 8b). Characters are also well grouped into text lines. In addition inclusions of characters are separately grouped, e.g. for the words "Grundlagen", "Konzepte", "Auflage", "Addision". Compositions of regions arise also for the image and the multicolored logo. Such non-text candidates are subsequently rejected by fusing the results of both methods.

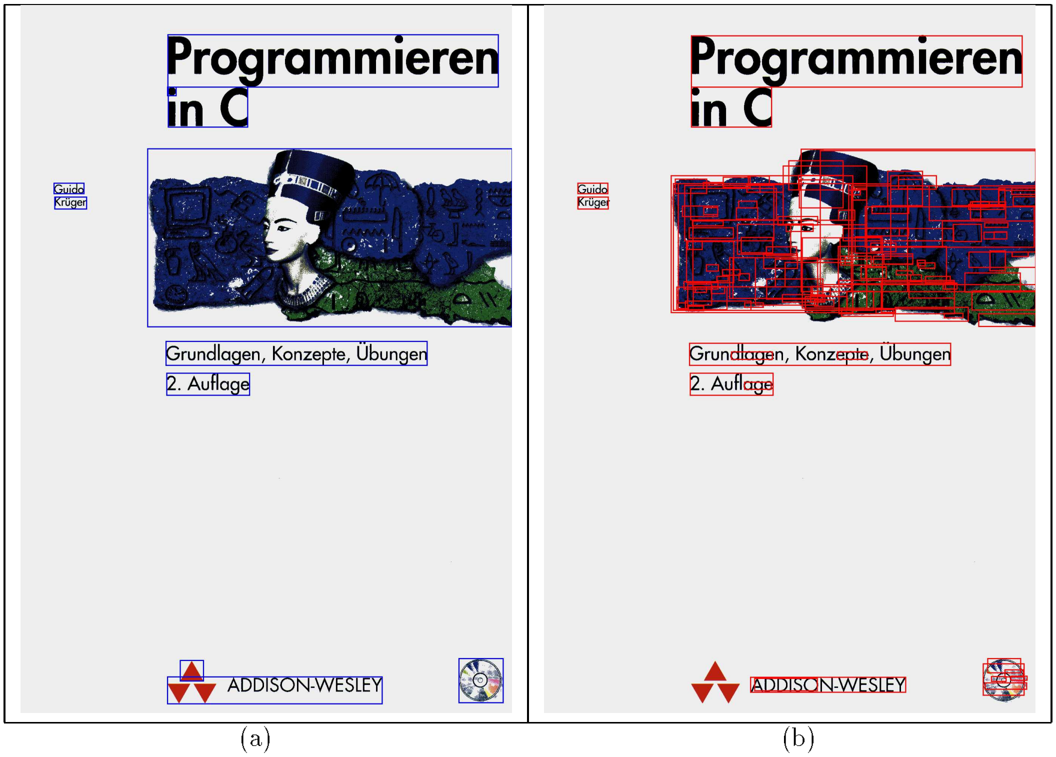


Figure 8: Result of composition of regions determined by (a) top-down analysis and (b) bottom-up analysis

7 Text Identification and Binarization

The definite decision whether a region contains text or non-text is difficult. In our approach we make use of the fact that the bottom-up and top-down analysis compute text candidates independently of each other, and combine the results of both. Identified text regions are binarized by using automatically extracted information about the text color.

Results of bottom-up and top-down analysis are combined by comparing the text candidates from one region to another. In the trivial case, one text candidate of bottom-up analysis is more or less the same as one text candidate of top-down analysis. Since both methods vote for text, the region is identified as text. The text color is stored for binarization. If the two text candidates differ considerably from each other, e.g. in size or text color, the regions are rejected as non-text.

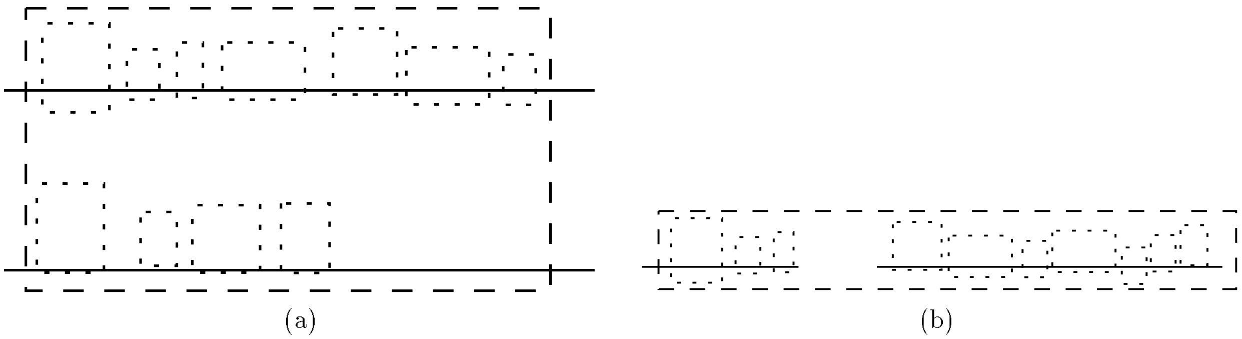


Figure 9: Example in which one text candidate of top-down analysis (dashed lines) corresponds to several text candidates of bottom-up analysis (dotted lines): (a) vertically aligned text candidates (b) horizontally aligned text candidates

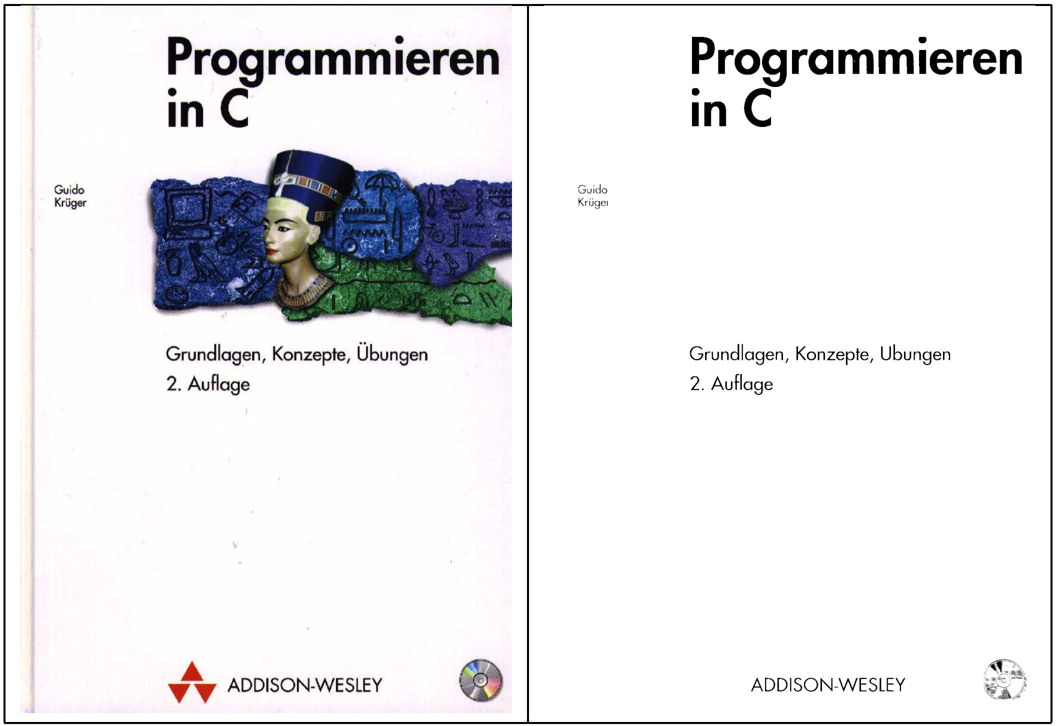
In case one text candidate of top-down analysis corresponds to several text candidates of bottom-up analysis, further analysis is necessary. It is checked whether the text candidate found by the top-down analysis corresponds to several lines of text, or several words on the same line of text reported by the bottom-up procedure. The first case is checked by the extraction and evaluation of base lines. For each text candidate reported by the bottom-up analysis, a base line is computed. It is assumed at the position where the mean value of the lower border of bounding boxes of these regions occurs (Fig. 9a). If the vertical distance of the base lines is larger than the mean height of the text candidates and all text candidates belong to the same cluster, the text candidates are identified as text. Otherwise it is checked whether the text candidates of bottom-up analysis are aligned horizontally (Fig. 9b). If several text candidates can be found that have the same base line and text color, they are identified as text, too. Text candidates that cannot be grouped are rejected as non-text. The case that one text candidate of bottom-up analysis corresponds to several text candidates of top-down analysis cannot arise due to the characteristics of the bottom-up and top-down procedures.

After text regions are identified they are binarized using the automatically extracted text color. For each pixel of a text region it is checked whether its color corresponds to the text color or not. All text color pixels are set to black in the binarized image. The binarized image obtained thus can be used as input for a conventional OCR module.

8 Results on Test Data Set

Our test data base consists of 16 book and journal covers of different complexity. It contains examples with homogeneous, multicolored and textured background. In some examples there are even background images. The appearance of text varies in terms of color, font, size and style. The book covers were scanned with a resolution of 200 dpi and a color depth of 24 bit. Test images have a typically size of about 1300 x 1800 pixels and include thousands of colors. Examples for results are shown in Figs. 10–12. Similar results were achieved on the other images of the test data base.

In Fig. 10a the scanned image of a book cover containing a complex image, several text elements and logos is shown. The result of text identification and binarization is given in Fig. 10b. As can be seen all



Programmieren in C

Guido Krüger

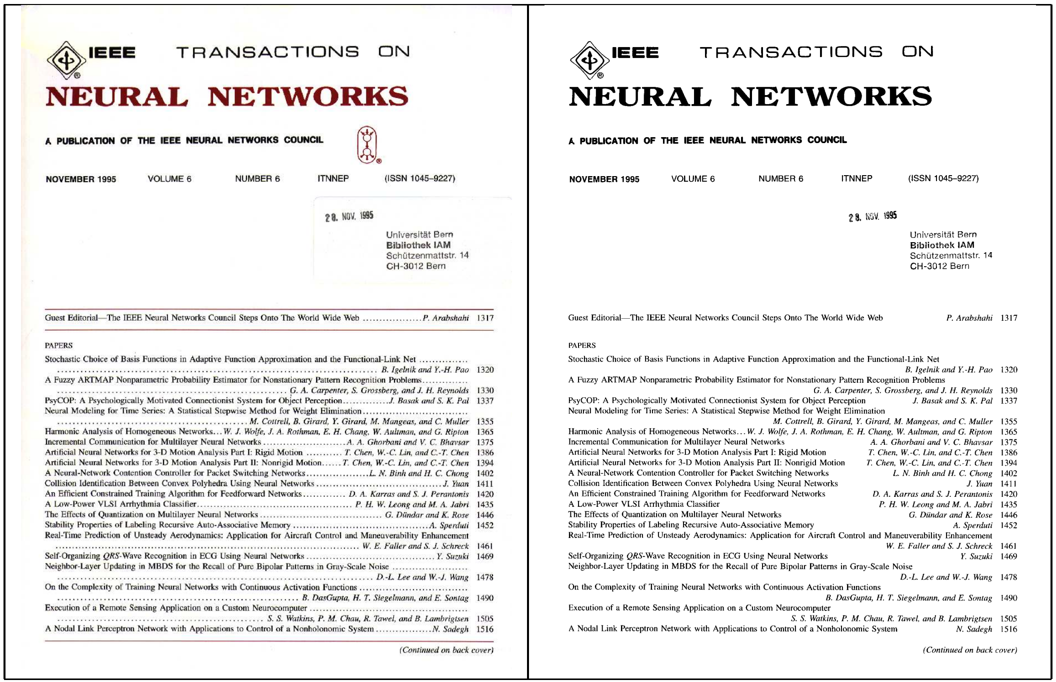
Grundlagen, Konzepte, Übungen
2. Auflage

ADDISON-WESLEY

(a) (b)

Figure 10: Result of text identification: (a) original color image (b) binarized text elements

text regions are well detected and binarized. The image is correctly rejected as non-text region. The logo of the publisher Addision-Wesley is also classified as non-text. A false alarm occurs for the CD-logo in the bottom right corner. However, it can be expected that this logo will be rejected by an OCR module.



IEEE TRANSACTIONS ON
NEURAL NETWORKS

A PUBLICATION OF THE IEEE NEURAL NETWORKS COUNCIL

NOVEMBER 1995 VOLUME 6 NUMBER 6 ITNNEP (ISSN 1045–9227)

28. NOV. 1995

Universität Bern
Bibliothek IAM
Schützenmattstr. 14
CH-3012 Bern

Guest Editorial—The IEEE Neural Networks Council Steps Onto The World Wide Web *P. Arabshahi* 1317

PAPERS

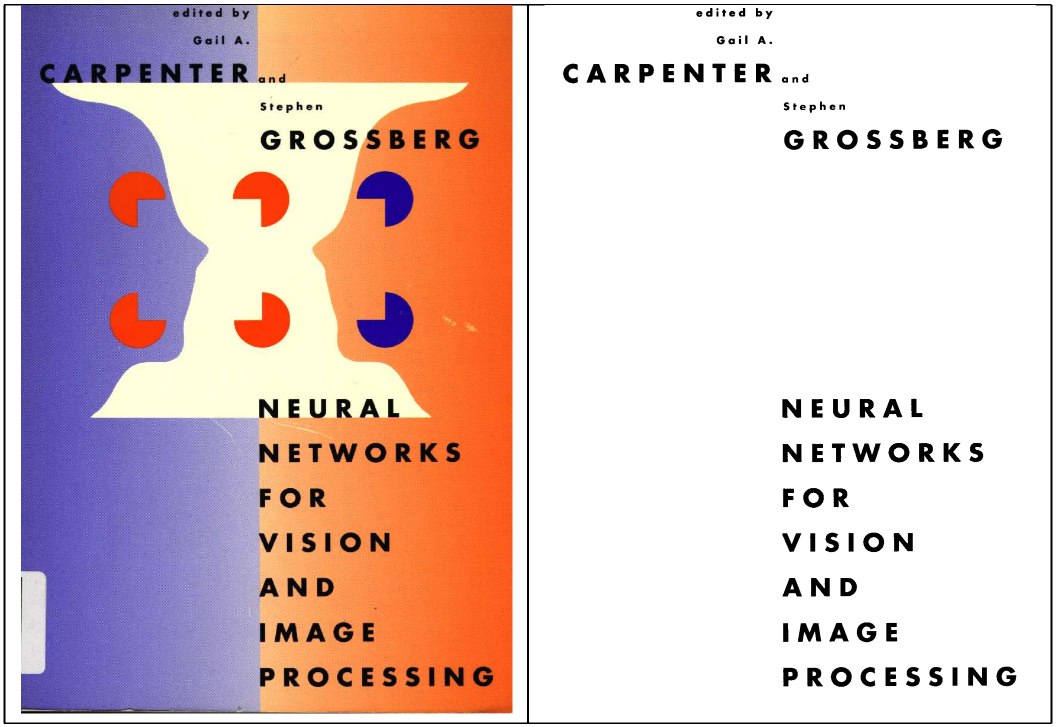
Stochastic Choice of Basis Functions in Adaptive Function Approximation and the Functional-Link Net *B. Igelnik and Y.-H. Pao* 1320
A Fuzzy ARTMAP Nonparametric Probability Estimator for Nonstationary Pattern Recognition Problems *G. A. Carpenter, S. Grossberg, and J. H. Reynolds* 1330
PsyCOP: A Psychologically Motivated Connectionist System for Object Perception *J. Basak and S. K. Pal* 1337
Neural Modeling for Time Series: A Statistical Stepwise Method for Weight Elimination *M. Cottrell, B. Girard, Y. Girard, M. Mangeas, and C. Muller* 1355
Harmonic Analysis of Homogeneous Networks... *W. J. Wolfe, J. A. Rothman, E. H. Chang, W. Aultman, and G. Ripton* 1365
Incremental Communication for Multilayer Neural Networks *A. A. Ghorbani and V. C. Bhavsar* 1375
Artificial Neural Networks for 3-D Motion Analysis Part I: Rigid Motion *T. Chen, W.-C. Lin, and C.-T. Chen* 1386
Artificial Neural Networks for 3-D Motion Analysis Part II: Nonrigid Motion *T. Chen, W.-C. Lin, and C.-T. Chen* 1394
A Neural-Network Contention Controller for Packet Switching Networks *L. N. Binh and H. C. Chong* 1402
Collision Identification Between Convex Polyhedra Using Neural Networks *J. Yuan* 1411
An Efficient Constrained Training Algorithm for Feedforward Networks *D. A. Karras and S. J. Perantonis* 1420
A Low-Power VLSI Arrhythmia Classifier *P. H. W. Leong and M. A. Jabri* 1435
The Effects of Quantization on Multilayer Neural Networks *G. Dündar and K. Rose* 1446
Stability Properties of Labeling Recursive Auto-Associative Memory *A. Sperduti* 1452
Real-Time Prediction of Unsteady Aerodynamics: Application for Aircraft Control and Maneuverability Enhancement *W. E. Faller and S. J. Schreck* 1461
Self-Organizing *QRS*-Wave Recognition in ECG Using Neural Networks *Y. Suzuki* 1469
Neighbor-Layer Updating in MBDS for the Recall of Pure Bipolar Patterns in Gray-Scale Noise *D.-L. Lee and W.-J. Wang* 1478
On the Complexity of Training Neural Networks with Continuous Activation Functions *B. DasGupta, H. T. Siegelmann, and E. Sontag* 1490
Execution of a Remote Sensing Application on a Custom Neurocomputer *S. S. Watkins, P. M. Chau, R. Tawel, and B. Lambrigtsen* 1505
A Nodal Link Perceptron Network with Applications to Control of a Nonholonomic System *N. Sadegh* 1516

(Continued on back cover)

(a) (b)

Figure 11: Result of text identification: (a) original color image (b) binarized text elements

Results for a second example are shown in Fig. 11. In Fig. 11a the original scanned journal cover is shown. It contains text printed in two different colors, several logos, lines, and a stamp of the library. As can be seen in Fig. 11b the small sized text is well identified and binarized. Also the lines and one of the logos are correctly rejected as non-text. The IEEE logo is erroneously grouped with the text of the first line and thus not rejected. Note that even the library stamp is correctly identified as text although it is not perfectly horizontally aligned.



(a) (b)

Figure 12: Result of text identification: (a) original color image (b) binarized text elements

Results for a book cover with a complex background image are shown in Fig. 12. Although text lines are printed on multicolored background, text components are correctly identified and binarized, and the background image is rejected as non-text.

9 Conclusion

In this paper an approach to automatic text identification and binarization on colored book and journal covers is proposed. Of course, the approach can also be applied to other kinds of color images, such as Web images or video images.

To reduce the large number of variations in color that arise due to digitalization, a graph-theoretical color clustering algorithm is applied in a preprocessing step. Afterwards text candidates are located using a top-down analysis based on successive splitting and a bottom-up analysis based on region growing. In order to find subsets of regions which are aligned in lines, a grouping step is applied next. Finally text regions and non-text regions are distinguished by comparing the results of both methods. Identified text regions are binarized by using automatically extracted text colors. With the proposed approach we obtained promising results on the images of our test data base. Future work will include the integration of an OCR module. Using feedback information from this module, a further increase in robustness of text identification is expected.

References

[1] I. Aksak, Ch. Feist, V. Kiiko, R. Knoefel, V. Matsello, V. Oganovskij, M. Schlesinger, D. Schlesinger, and G. Stanke. Extraction of filled-in data from colour forms. *Lecture Notes in Computer Science (LNCS)*, 1296:98–105, September 1997.

[2] O. Déforges and D. Barba. A fast multiresolution text-line and non text-line structures extraction and discrimination scheme for document image analysis. In *ICIP Proceedings 1994*, volume 1, pages 134–138, August 1994.

[3] A. K. Jain and B. Yu. Automatic text location in images and video frames. Technical Report, Michigan State University, September 1997.

[4] Y. Liu, T. Yamamura, N. Ohnishi, and N. Sugie. Detecting characters in grey-scale scene images. *Lecture Notes in Computer Science (LNCS)*, 1352:153–160, January 1998.

[5] J. Matas and J. Kittler. Spatial and feature space clustering: applications in image analysis. In *Proceedings of the 6th Int. Conf. on Computer Analysis of Images and Patterns*, pages 162–173. Springer, September 1995.

[6] T. Sato, T. Kanade, E. K. Hughes, and M. A. Smith. Video OCR for digital news archive. *IEEE International Workshop on Content-Based Access of Image and Video Database*, pages 52–60, January 3 1998.

[7] Y. Zhong, K. Karu, and A. K. Jain. Locating text in complex color images. *Pattern Recognition*, 28(10):1523–1535, October 1995.

[8] J. Zhou and D. Lopresti. Extracting text from WWW images. In *Fourth International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 248–252, August 1997.