Character Recognition in Bookshelf Images using Context-based Image Templates

Minako Sawaki, Hiroshi Murase and Norihiro Hagita
{minako, murase, hagita}@apollo3.brl.ntt.co.jp
NTT Communication Science Laboratories
3-1, Morinosato-Wakamiya, Atsugi, Kanagawa, 243-0198, Japan

Abstract

This paper proposes a method for recognizing degraded characters in bookshelf images captured by a digital camera. We adopt displacement matching and templates that include neighboring characters or parts thereof to cope with the degradation. The templates are referred as context-based image templates, since they offer more contextual information than single-letter templates. Such templates are effective where ever there is a restricted word set, such as journal titles, year, month, volume, and number. Experiments with 3,468 characters in nine bookshelf images show that this method achieves a higher recognition rate (96.3%) than single-letter templates (88.4%).

1. Introduction

A digital camera is currently one of the most inexpensive and simple tools for gathering image data. It may also be extended to a convenient tool for image-to-text code conversion. We attempt to recognize the characters on the spines of technical journals on bookshelves using digital camera images. The goal is to construct a personal library/filing database of the journals in the user's vicinity.

Characters in bookshelf images (Fig. 1) are often degraded because of the poor printing and viewing conditions. This degradation problem involves problems with computer vision as well as conventional character recognition. To cope with the degradation problem, most conventional methods for scene images like bookshelf images consist of two stages: character extraction and character recognition. They try to extract characters based on character color and/or complexity as in [1,2], in order to utilize existing character recognition methods in the second stage. However, when images include noise or are otherwise degraded, they are rarely extracted with sufficient accuracy. For example, when the spine is wrinkled or scratched, it is difficult to separate individual characters from the text-line in preprocessing.

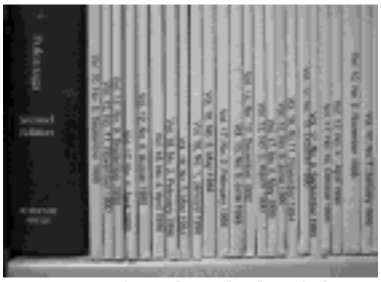

Fig. 1 Example of a bookshelf image.

We employ the multiple-template method and displacement matching to solve this problem. Displacement matching is a recognition-based segmentation method and reduces segmentation errors [3]. For displacement matching, the complementary similarity measure [4] is very robust against deletion and additive noise as the discriminant function. However, even with the complementary similarity measure, recognition performance for small characters are still low. We, therefore, introduce context-based image templates for improving the recognition accuracy. The context-based image templates include not only the target character but also its neighboring characters or parts thereof. Since the context-based image templates express a larger area, they are effective in improving the recognition accuracy. In other words, the context-based image templates represent more contextual information in terms of the image pattern space than the single-letter templates. This yields different properties from error correction methods with the conventional *n*-gram of categories in postprocessing after recognition. The postprocessing method, in general, requires high recognition rates for individual character recognition. In our problem, the postprocessing may not be available due to the lower recognition rate for small characters. Therefore, our approach is also effective against degraded characters.

2. Recognition process (Fig. 2)

The input color image is converted into a gray-scale image and then binarized. Skew detection is needed since journals usually slump on the bookshelf. Boundaries of journals are detected based on the dark lines caused by shadow. Next, text-line regions between the boundaries are extracted by the transition frequency of pixel colors (white-to-black and black-to-white) during vertical scanning, and then the skew of each text-line is corrected .

Characters in the text-line region are recognized by displacement matching [4]. An observation window ***F*** moves along the text-line, and ***F*** is matched against stored templates. ***F*** is also shifted along the *x*-axis to cover text-line misalignment. The complementary similarity measure, which is robust against noise, is employed for matching [4]. If the maximum similarity value exceeds a threshold, the category is determined to be a recognition result.

The appropriate window shape is a matter to be examined further, but we utilize a uniform square window in this paper. When performing displacement matching, characters adjoining the target character are included in ***F***. These neighboring characters are usually noise and may be eliminated if the window size can be changed to match the category. However, if the window shape is changed according to a category, small characters (like '*l*', '*1*') in a small window are often completely matched to a portion of large characters (like '*M*', '*N*') and take high similarity values. As a result, almost all recognition results are occupied by such small characters and large characters are seldom recognized correctly. Moreover, a square window can contain at least a target character, while it seldom contains second previous or second next characters of the target character. Since the second previous or second next characters have less effect on the target character with regard to pattern shape of the target character, they are not expected to be included in ***F***.

3. Context-based image templates

3.1 Basic framework

In order to recognize degraded characters accurately, we employ information about the source materials. Specific combinations of some categories (like "*ber*") are often observed in the titles of journals, and these combinations are used to create context-based image templates, ***Tc***, that include multiple characters (a target character and parts of its neighbors). The conventional templates that include only one character are called single-letter templates ***Ts***. ***Tc*** carry more image information than ***Ts*** and so are better suited to degraded character recognition. In ***Tc***, contextual information is buried into the pattern itself. Contextual information is conventionally used in postprocessing after recognition, however, we employ this information in the recognition stage. By utilizing contextual information from neighboring characters for recognition, the noise problem caused by neighboring characters in a uniform window can also be solved.

When the vocabulary of a task is limited, the number of combinations of consecutive categories is limited. Therefore, template number is not excessive even if templates consisting of three categories are used. Template number can be minimized by employing the selection of effective templates as will be mentioned later.

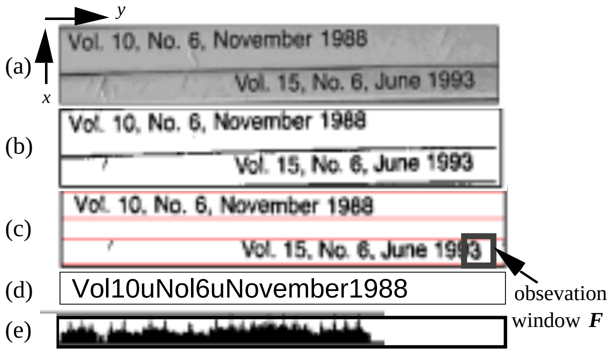

Fig. 2 Recognition process.
(a) gray-scale image, (b) binary image, (c) skew corrected text-lines, (d) recognition results of the upper line, (e) maximum similarity values of the upper line.

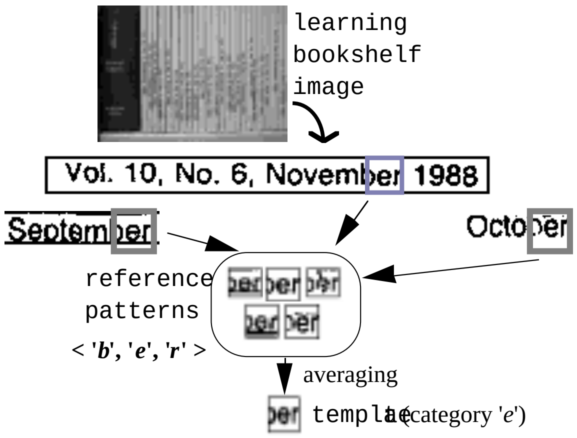

Fig. 3 Creation of context-based image templates.

Another approach to employing contextual information is to use word templates. However, this approach requires all word templates to be recognized, so matching time is excessive.

3.2 Creation of context-based image templates

From learning bookshelf images, text-lines are extracted through the preprocessing mentioned in section 2.1. For all text-lines, character positions are indicated manually. As reference patterns, square regions of constant size, each of which includes a target character at their centers, are extracted from the text-lines. The reference patterns are labeled with the sequence of the categories of the three characters (the previous, the target, and the next character). In order to reduce the variation of labels, we use labels with three categories even though some reference patterns include more than three characters. All reference patterns of the same label are averaged and binarized. Averaging reduces template number and removes noise pixels. The obtained binary pattern is stored as context-based template

Tc (Fig. 3).

3.3 Selection of context-based image templates

One problem with multiple templates is that the processing time increases in proportion to the number of templates. Therefore, a method for selecting only effective context-based image templates ***Tc*** is needed to reduce the template number. In this paper, we prepare single-letter templates ***Ts*** for all categories and ***Tc*** of selected categories. ***Tc*** are to recognize input patterns of specified combinations of categories; ***Ts*** are to recognize other unregistered combinations.

The selection of ***Tc*** is based on the recognition accuracy achieved by ***Ts***. We select only those ***Tc*** whose categories suffer low recognition accuracy if only ***Ts*** are used. This means that the selected templates are for small characters only. Large character categories can be adequately recognized using ***Ts***.

To balance the similarity values of ***Tc*** and ***Ts***, the similarity values for ***Ts*** are weighted. As the similarity values of ***Ts*** tend to be smaller than those of ***Tc***, categories with ***Ts*** only are rarely recognized without weighing. The value of the complementary similarity measure S_c lies in the range

$$-\sqrt{T(n-T)} \# S_c \# \sqrt{T(n-T)} \qquad (1)$$

where T is the number of black pixels of a template and n is the total pixel number of a template. When T is smaller than $n/2$, the larger is the black pixel number, the larger is the upper bound of the similarity. Usually, T of character templates is smaller than $n/2$, so the upper bound of ***Tc***, is larger than that of ***Ts***. Consequently, for large character categories with ***Ts***, similarity values tend to be smaller.

We define the weight so that the upper bound of ***Ts*** equals that of ***Tc***. Weight $w(c)$ ($w(c)$: weight value for category c) for ***Ts*** is defined as follows. At first, the averaged black pixel number of all reference patterns of c is calculated and the upper bound $Uc(c)$ of the similarity value is obtained from *eq.* (1). Next, the black pixel of ***Ts*** of the same c is obtained and the upper bound $Us(c)$ of the similarity value is calculated from *eq.* (1). The ratio of these ($Uc(c)$ / $Us(c)$) is defined as weight $w(c)$ for c.

In displacement matching, when matching to ***Ts***, weight $w(c)$ is multiplied by the complementary similarity value and the resultant value is used as the new similarity value. This weighing prevents ***Ts*** from having low similarity values and maintains the recognition accuracy for both categories with and without ***Tc***.

4. Recognition experiments

4.1 Experimental conditions

Bookshelf images were captured by using a digital camera (832x608 pixels) from a straight view. Journals on the bookshelf were the same kind with different publication dates (journals : *IEEE Trans. Pattern Anal. Machine Intell.*, 1988-1996). Threshold value for binarization was 128 out of 256 gray levels.

Thirty seven categories were used consisting of numbers (*0-9*) and alphabetical categories (*A,D,F,J,M-O,S,V,a-c,e,g-i,l-p,r-v,y*) which are enough for recognizing *No*, *Vol*, *month* and *year*, all of which are often printed on journal spines. Dots (.) and commas (,) were not recognized. 3,845 patterns in 10 bookshelf images and 3,468 patterns in 9 bookshelf images were used as reference and test patterns, respectively (both had 14 words and numbers). The size of ***F*** was n = 24x24 pixels.

4.2 Experimental results

Total number of templates was 173, including 136 context-based image templates ***Tc*** (1 - 18 pat./cat.), and 37 single-letter templates ***Ts*** (1 pat./cat.). ***Ts*** were obtained as follows. Initial single-letter reference patterns were extracted manually from one reference pattern per category. Next, neighboring characters in ***Tc*** were eliminated automatically by deleting everything outside the rectangle circumscribing the initial single-letter reference patterns of the same category. The resulting patterns were averaged and used as ***Ts***.

The recognition results for the test patterns are shown in Table 1. In the experiments, the result was regarded as correct when the correct category was determined at the correct position. Table 1 shows that the proposed method achieved the recognition rate of 96.3%. Two main reasons for recognition errors with the proposed method were pattern deformation caused by low-quality printing or preprocessing and thresholding. For comparison, templates without contextual information were made from ***Tc*** and used for recognition. Neighboring characters in ***Tc*** were eliminated by deleting everything outside the rectangle circumscribing the initial single-letter reference pattern of the same category. The recognition rate was 88.4%. This shows that the proposed method achieves much higher recognition rates than the conventional one even with the same number of templates. As an another conventional method, only ***Ts*** were used for recognition. The recognition rate was 77.1%.

Recognition rates of the small characters of '*l*' and '*r*' were examined with ***Tc***, and templates without contextual information, in both cases 173 templates were used. The number of error patterns was decreased from 47 to 6 in 223 patterns for category '*l*' and from 75 to 21 in 146 patterns for category '*r*' by utilizing contextual information. For all capital categories, the number of error patterns decreased from 16 to 3 for 549 patterns. These results show that the error reduction was most noticeable for the small characters.

The effects of appending selective context-based image templates were investigated (Fig. 4). In the experiments, context-based image templates were added to ***Ts*** starting from the narrow character categories and moving to the wide character categories. The results shows that the recognition rates increase as the context-based image templates are added. The effect of the contextual information is about 10% and depends only slightly on the number of templates. The 20% increase in the recognition rates by adding all context-based image templates is due to two reasons: the effect of neighboring characters and the effect of the increase in template number. Half of the 20% can be obtained with just 30 templates of '*1*', '*i*', '*l*', '*r*' and '*t*', all small characters.

The recognition accuracy for different window shapes was also investigated. The window shape was determined for each category as the circumscribed rectangle of the initial single-letter reference pattern of the corresponding category. The four-fold correlation coefficient was used as the discriminant function to normalize the similarity values for different window sizes. The recognition rate was 40.2%. 65.6% of the recognition results was occupied by category '*l*', although only 6.4% of the input test patterns involved '*l*'. This shows that changing window shape according to the category does not increase the performance for this task.

Table 1 Recognition rates (test data).

	context-based image templates	templates without contextual information
recognition rates (%)	96.3	88.4

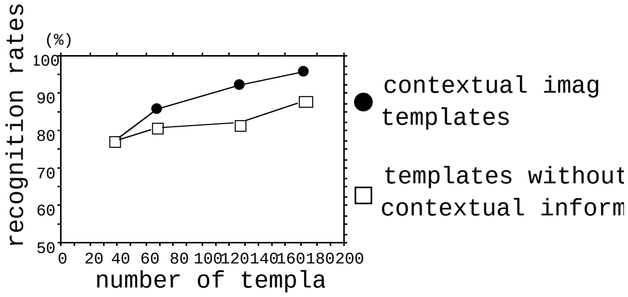

Fig. 4 Recognition rates versus template number (test data).

5. Journal volume retrieval system

As an application of the proposed method, a journal volume retrieval system was constructed. This system enables us to locate the desired volume within similar magazines on the bookshelf or to determine if some volumes are missing.

This system consists of a digital camera for image capturing and a computer for recognition and retrieval, both commercial products. It recognizes characters in bookshelf images, and search strings are matched against the recognition results. When the desired journal volumes exist, they are displayed on the screen as binary images. Two search strings can be typed at one time, and the system retrieves the volumes using AND search.

The system image is shown in Fig. 5. The recognition results from top to the bottom correspond to the magazines from left to right. In Fig. 5, two journals are retrieved using the search strings of "*Nov*" and "*11*".

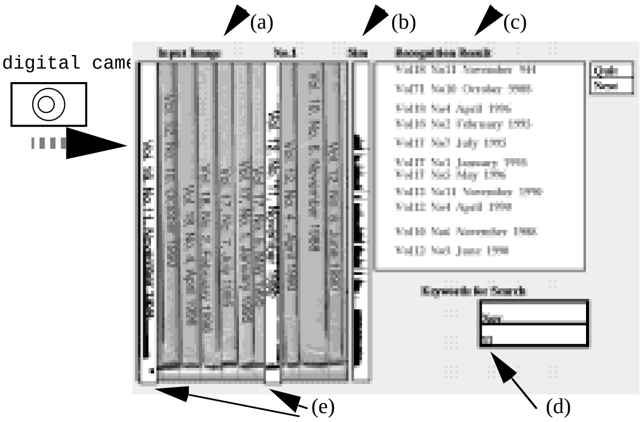

Fig. 5 Journal volume retrieval system
(a) input color image, (b) maximum similarity values, (c) recognition results, (d) search strings, (e) retrieved volumes.

6. Conclusions

This paper proposed a method for recognizing characters in bookshelf images captured by a digital camera. We employed displacement matching and context-based image templates of frequently occurring character sequences. The context-based image templates represent more contextual information than single-letter templates and so offer more robust recognition under heavy noise. Experiments on bookshelf images show that the proposed method achieves a higher recognition rate (96.3%) than single-letter templates (88.4%). As an application of the proposed method, a journal volume retrieval system was constructed.

Our future works include the automatic extraction of reference patterns and applying this method to various kinds of books.

Acknowledgements
We wish to express our gratitude to Dr. K. Ishii, Mr. B. Maire and Mr. T. Oda for their cooperation.

References
[1] J. Ohya, A. Shio, and S. Akamatsu, "Recognizing Character in Scene Image", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 16, No. 2, pp. 214-220 (1994).
[2] S. Kurakake, H. Kuwano, H. Arai, and K. Odaka, "Key target indexing by recognition for content-based retrieval", *Tech. Report of IEICE*, PRU95-237, pp. 15-20 (1996) (in Japanese).
[3] R. G. Casey and E. Lecolint, "A Survey of Methods and Strategies in Character Segmentation", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 18, No. 7, pp. 690-706 (1996).
[4] M. Sawaki and N. Hagita, "Text-line Extraction and Character Recognition of Document Headlines with Graphical Designs using Complementary Similarity Measure", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 20, No. 20, pp. 1103 - 1109 (1998).