

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI[®]

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

AUTOMATIC CLASSIFICATION OF MULTI-LINGUAL
DOCUMENTS

JIE DING

A THESIS
IN
THE DEPARTMENT
OF
COMPUTER SCIENCE

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE
CONCORDIA UNIVERSITY
MONTRÉAL, QUÉBEC, CANADA

APRIL 1999
© JIE DING, 1999



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-39111-6

Abstract

Automatic Classification of Multi-lingual Documents

Jie Ding

Language classification (LC) refers to the categorization of text documents into different natural language groups, whereas language identification (LI) determines the language used in a document. LC and LI play important roles in document processing systems, because they can perform initial classifications to reduce the scope for subsequent stages of processing. Two major parts of the work are: (1) LC of documents written in 24 languages into two language categories (oriental and European), and (2) LI of oriental documents into Chinese, Japanese and Korean.

This thesis concentrates on the exploration of statistical features that can contribute to LC/LI, as well as the design and implementation of programs to differentiate between documents printed in various natural languages. A total of six distinctive features are proposed and used in this study.

For LC, three features are used: horizontal projection profiles, height distributions of connected components (CC) and enclosing structure of connected components. Experimental results show that we are able to classify the script of a document as either European or Asian based on four 50-CCs and obtain a high recognition rate while maintaining a low rejection rate.

In the LI of oriental documents, the complexity of structure, Korean “circles” and vertical strokes have been chosen for distinguishing features between the three language scripts. The identification has been made according to the range of values in these features, and also by K-means clustering.

When applied to seven hundred documents in the CENPARMI Lab, the recognition rates achieved in LC and LI have exceeded 95% and 94%, with error rates that are below 2% and 4.5%, respectively.

Acknowledgments

First of all, I would like to extend my sincere gratitude to both of my supervisors Drs. C.Y. Suen and Louisa Lam for their guidance, encouragement, and help throughout my studying in Concordia. The time and effort they dedicated to the project and my thesis are crucial and indispensable.

I am also grateful to people whose studies have directly contributed to this project. Among them are Nicholas Strathy whose image processing code libraries were used in my project, Drs. Ke Liu and Didier Guillevic who both made helpful suggestions to different aspects of this project, Christine Nadal and Guiling Guo who helped collecting the sample documents, William Wong and Mike Yu who provided effective technical support. Also I very much appreciated Dr. Y. Y. Tang's advice.

Special thanks should go to my colleagues and fellow graduate students Hao Chen, Rong Fan, Eddie Webb, Jie Zhou, Qizhi Xu, Xiangyun Ye and Boulos Waked who have given me their help at different times and in different ways since I first joined CENPARMI.

Finally, a secret source of my confidence, courage, and strength should be mentioned. It's my husband Yuan's love, understanding and support that enabled me to complete this thesis.

Contents

List of Figures	vii
List of Tables	ix
1 Introduction	1
1.1 Review and Related Work	2
1.1.1 Differentiation between Roman and Oriental Languages	3
1.1.2 Classifying of Chinese, Japanese and Korean Scripts	4
1.2 Introduction of Our Work	6
1.3 Organization of this Thesis	8
2 Preprocessing	9
2.1 Noise Removal	9
2.2 Segmentation	9
2.3 Deskewing	10
2.4 Line Concatenation	12
2.5 Size Normalization	13
3 Differentiation between European and Oriental Scripts	14
3.1 Introduction	15
3.2 Work Based on Concavity Feature	16
3.3 Feature Description	19
3.3.1 Horizontal Projection Profile	20
3.3.2 Height Distribution of Connected Components	24
3.3.3 Bounding Box Overlapping/Enclosing	30
3.4 Combination of the Features	31
3.5 Experimental Results	32

3.5.1	Datasets	32
3.5.2	Results	32
3.5.3	Analysis of Results	34
4	Identification of Chinese, Japanese and Korean	37
4.1	Characteristics of Chinese, Japanese and Korean Characters	37
4.2	Work Based on Optical Density	39
4.3	Feature Extraction	42
4.3.1	Complexity of Structure	42
4.3.2	Korean “circles” and “ellipses”	43
4.3.3	Korean Vertical Strokes	49
4.4	Identification	55
4.4.1	Feature Vector Construction	55
4.4.2	Identification	55
4.4.3	Rejection Criteria	64
4.5	Experimental Results	64
4.5.1	Classification Results according to C, K, V values	64
4.5.2	Classification Results from Clustering	64
4.5.3	Comparison of Clustering Results from Using Two Features	66
4.5.4	Analysis of Results	66
5	Summary and Future Work	71
5.1	Summary	71
5.2	Main Contributions of the Thesis	72
5.2.1	Contribution 1: New Features to Separate European and Oriental Documents	73
5.2.2	Contribution 2: New Features to Identify Chinese, Japanese and Korean Documents	73
5.3	Conclusion and Future Work	74
5.3.1	Conclusion	74
5.3.2	Future Work	74

List of Figures

1	Examples illustrating average number of runs in a cell	6
2	Horizontal, Vertical Smoothing, AND Operation and Segmentation Result	12
3	(1) Two Text Lines (2) Concatenation along the bottom of bounding box (3) Concatenation along the baseline	13
4	Upward Concavity of Letter V	16
5	Reference lines separating European characters into three zones . . .	16
6	Upward concavity distribution of several text-lines	17
7	(1) A Swedish text line (2) A Russian text line (3) Concavity histograms	20
8	Several horizontal projection profiles	21
9	A horizontal projection profile of a Bulgarian script line	23
10	A European text line and its connected components representation . .	25
11	A Chinese text line and its connected components representation . . .	25
12	Height distribution of the components of oriental languages	26
13	Height distribution of the components of European languages	27
14	A European text line and its normalized height distribution histogram	29
15	A Sample Chinese Character with Enclosed Structure	30
16	A French Line Misclassified as Oriental	35
17	A Swedish Line Misclassified as Oriental	35
18	A German Line Misclassified as Oriental	35
19	A German Line Misclassified as Oriental	36
20	A Chinese Line Misclassified as European	36
21	A Japanese Line Misclassified as European	36
22	A Korean Line Misclassified as European	36
23	A Chinese Line Misclassified as European	36

24	Method of Lee et al. when applied to Chinese and Japanese training samples	40
25	When applied to Chinese, Japanese and Korean training samples . . .	41
26	Character cells of a Japanese text line	42
27	Examples of complex structure	42
28	Characters with Complex structure in a Chinese Text Line	43
29	Examples of characters containing short and tall loops	44
30	Six structures of Korean characters	45
31	Loop's external contour containing other parts	46
32	A Japanese text line	46
33	A Korean text line	47
34	Rectangle contained in a Korean text line	49
35	100 Most Frequently Used Korean Characters	50
36	Korean text line illustrating different shapes of vertical strokes	52
37	Non-symmetry of Korean vertical stroke	54
38	Feature vectors of the three training sets	56
39	Chinese training data and their cluster centers	61
40	Japanese training data and their cluster centers	62
41	Korean training data and their cluster centers	63
42	Some Chinese Characters in Kai Font	69
43	Korean document containing ellipses	70
44	Korean document containing sharp turns in circles	70

List of Tables

1	The results of language classification by concavity location	19
2	The results of language classification by using one 50-component . . .	32
3	The results of language classification by using two 50-components . .	33
4	The results of language classification by using three 50-components .	33
5	The results of language classification by using four 50-components . .	33
6	Feature vectors of Chinese training samples	57
7	Feature vectors of Japanese training samples	58
8	Feature vectors of Korean training samples	59
9	Twelve cluster centers	61
10	Results of oriental language classification by using C, K and V values	65
11	Confusion matrix when using C, K and V values	65
12	Results from clustering using C, K and V features	65
13	Confusion matrix from clustering using C, K and V features	65
14	Results of clustering from using C and K features	67
15	Confusion matrix from using C and K features	67
16	Results of clustering from using K and V features	67
17	Confusion matrix from using K and V features	67
18	Results of clustering from using C and V features	68
19	Confusion matrix from using C and V features	68

Chapter 1

Introduction

Language is one of the most important tools for human communication. Today, hundreds of languages exist in the world [Cry94], [Nak80]. Given a document, identifying the language to which it belongs is both useful and interesting. This topic is referred to as *language identification*, or *language differentiation* in the pattern recognition area. In the past, language identification was normally done manually because the amount of documents to be processed was limited. One could often deduce the language written in a document from its country of origin. If not, language interpreters could be employed to examine the document in order to determine its language. Hence the attention paid to automatic language identification was much less than that of Optical Character Recognition (OCR).

However, with increasingly more documents stored and manipulated electronically, manual language identification performed by human beings becomes inadequate. In order to address this issue, and also to enhance the efficiency of OCR, developing automatic language identification methods becomes a necessity. For example, early determination of the language used in a document has implications in the selection of proper character recognition algorithm which will also greatly facilitate further processing, such as indexing and translation. This approach is especially useful as an initial component of systems for processing multilingual documents, for an automatic selection of the appropriate classifier to be used in subsequent processing of the document.

1.1 Review and Related Work

Compared with optical character recognition, research work and results of language identification are not abundant. In the area of language identification, several aspects have been studied:

- Category classification: the hundreds of languages in the world are grouped into different categories, such as: Roman, Cyrillic, Arabic, oriental, etc. For a given document, identifying the language category of this document is the major work of category classification. Recent work includes [Spi94], [LNB96], etc.
- Language differentiation within a certain language category: which includes Roman language differentiation [SS94], [NS93], [NBS97]; oriental language differentiation [Spi94], [LNB96], [SH98], etc.
- Direct identification: this approach does not perform category classification, while it identifies each input document as a specific language. Related work includes [HKTK97].

Existing identification methods are mainly based on two kinds of approaches:

- Statistics(Computation)-based analysis and identification: from a given document, extracting various features (such as those related to optical characteristic, stroke density, character complexity, etc.), analysing and processing feature data, identifying the language based on calculated values of those features. Examples include: [Spi94], [LNB96], [SH98], etc.
- Token matching (also called template matching): it first determines a set of language specific image tokens for each language, then searches for such tokens in the input document and finds the best match [HKTK97], [SS94], [NS93], [NBS97].

Previously, the first approach was considered to be applicable for gross classification while the second one was used more frequently for specific classification, owing to the following reasons:

- Documents of languages in the same language category usually have similar statistical characteristics and those in different categories possess noticeable differences. These differences can be revealed by using computation-based analysis

approach.

- Each language normally has its own language specific tokens. Using these tokens to identify documents in each of the languages within the same category was considered to be more feasible.

The first approach has a higher computational complexity, while the second one needs a large database to extract enough specific tokens and a large memory space to store these tokens.

More research work has been done in the differentiation of Roman-alphabet languages and many sources of information have been used: short words [Kul91]; n-grams of words [Bat92]; n-grams of characters [CT94]; diacritics and special characters [Bee88], [New87]; syllable characteristics [Mus65]; morphology and syntax [Aie91]; and character code shapes [NBS97], [SS94], [NS93].

It is only in the past several years that researchers have begun to study the differentiation between oriental languages, and their research has been mainly focused on the separation of Chinese, Japanese and Korean scripts. In comparison with Roman language differentiation, fewer methods have been proposed for oriental scripts because of their complexity and the very large character sets.

In the following sub-sections we will introduce the research work in areas of differentiation between Roman and oriental language categories, followed by the differentiation between the three oriental languages. The work described is closely related to on-going research at the Centre for Pattern Recognition & Machine Intelligence.

1.1.1 Differentiation between Roman and Oriental Languages

In [Spi94], Spitz proposed a system which can classify documents into seven languages: English, French, German, Russian, Japanese, Chinese and Korean. According to this system, the differentiation between oriental and Roman scripts is based on the different distributions of the vertical locations of upward concavities in the two classes of languages. This difference in distribution arises from a basic difference in structure between these two classes of scripts. According to [Spi94], Roman scripts are mostly composed of lower-case characters, and therefore these concavities tend to be accumulated along the baseline and the x-line. On the other hand, oriental scripts are composed of radicals that can be located at any height within the text

line, resulting in a more random distribution of upward concavities in these scripts. Experiments performed by Spitz indicates this method works well in identifying the language category of documents in these seven languages without having to analyze every image line in the input documents. However, this reliance on the vertical locations of upward concavities will not work for some other European languages, such as Bulgarian a Cyrillic script, in which a noticeable portion of upward concavities do not cluster near the x-line and baseline. In order to be able to identify documents in those Cyrillic scripts, we need to search for other classification feature(s).

Another important result in this area is [LNB96] in which Lee et al. described their methods which employ multiple features to separate documents in Chinese and Japanese from English, French, German, Italian and Spanish. His emphasis is on the processing of degraded document images, therefore four features were used: (a) position of concavities, (b) distribution of character height, (c) character bounding-box top and bottom profile, and (d) Spitz's optical density feature. This method needs to analyze all image lines in a document before making a decision on its language category.

We did not apply the method of [LNB96] because this work was published after we have completed our work on the differentiation between the two language categories.

1.1.2 Classifying of Chinese, Japanese and Korean Scripts

For the oriental scripts which include Chinese, Japanese and Korean. Spitz [Spi94] proposed a method using optical density to differentiate the three languages. The optical density D_i of a character cell is defined as the number of 'on' pixels of this cell divided by its area, which can be represented as follows:

$$D_i = B_i / (H * W_i) \quad (1)$$

Where B_i is the number of 'on' pixels in the i -th cell, W_i is the width of the i -th cell and H is the height of the text line to which the i -th cell belongs. He showed that the histograms of the distribution of the optical density of these three languages are different: Chinese has only one significant mode; Korean is distinctly bi-modal, with the low density mode smaller than the high density mode; while Japanese is also characterized as bi-modal, but the relative heights of the modes are reversed: the lower density mode is greater than the high density. A linear discriminant analysis

(LDA) is applied to the histogram to classify the three languages.

We may notice that this optical density of characters is dependent not only on the language to which the character belongs, but also on the font and printing style. For example, the optical density of characters in heiti font of Chinese is greater than that of characters of songti, fangsongti or kaiti fonts. Even the optical density of Katakana, Hiragana and Korean characters can be larger than that of Chinese characters if they are printed in dark black bold (such as title, subtitle, etc.) for emphasis. Because of this, the method of calculating the optical density becomes critical in the above method. Another factor which may also affect the accuracy is the character cell segmentor. Determining a correct character cell is not easy because of the existence of punctuations, the different sizes of Kanji, Japanese Katakana, Hiragana characters, the left-right, left-middle-right structures of characters and occasionally the touching of some adjacent characters.

Because of these reasons, Lee et al. [LNB96] modified the optical density definition as the fraction of the black area multiplied by the average number of vertical runs per cell (assuming the text line is oriented horizontally). The average number of runs in a cell is defined as the total number of runs in this cell divided by the width of the cell. For illustration, Figure 1 shows two cells of width 10 and height 6, both having the same number of black pixels. The left cell has a single stroke of 3 pixels thick, and the right cell has 3 strokes, each of 1 pixel thick. The average number of vertical runs in the left cell is 1 ($10/10 = 1$), and 3 ($30/10 = 3$) for the right cell. This definition considers pure optical density as well as the difference introduced by font design. The density feature is calculated for all cells in a page as a distribution, then the linear discriminant is applied to the mean and variance of this distribution. As this method was only applied to separate Chinese from Japanese, its performance needs to be further investigated if we also need to differentiate Korean.

As pointed out in [SH98], this definition of optical density is still influenced by stroke width, and it uses the average number of runs that cannot be calculated in each cell. The authors of [SH98] thought the segmentation method of [LNB96] makes densities insensitive, because adjacent complex and simple characters are mixed into a cell. As density depends on character segmentation and complexities in oriental languages change considerably from character to character, it is very crucial to have an almost successful cell segmentor in order to apply the optical density. As thinned images are insensitive to stroke width, the authors proposed using contour density to

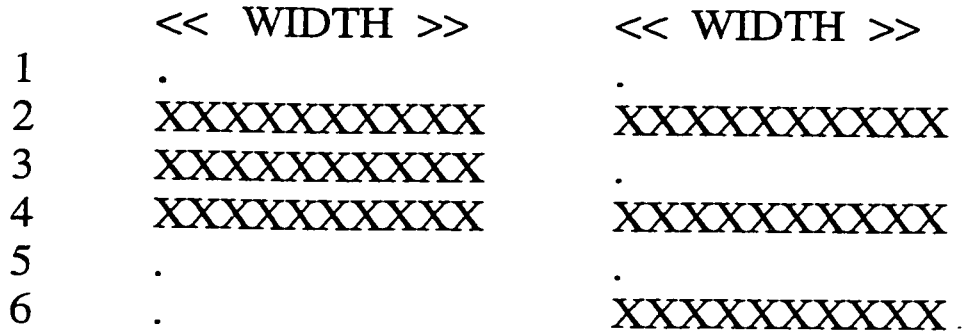


Figure 1: Examples illustrating average number of runs in a cell

obtain a density which is insensitive to stroke width while reducing the processing time required by thinning. However, this method was only experimented on 6 Chinese, 6 Japanese and 6 Korean sample documents.

1.2 Introduction of Our Work

It is worthwhile to mention that this thesis is dedicated to a research topic rather than a practical project which can be completed by using mature techniques. The task of this thesis is to explore an alternative and better method to solve a problem that is currently still open for study and experiments.

More specifically, this thesis presents an effort to address the language classification problem with scanned and stored documents. Our work deals with documents in three different language categories that involve more than 23 languages.

Two major aspects of this work have been conducted and described in this thesis:

- Language-category classification of documents from European and oriental sources. The European source is comprised of Roman languages and Cyrillic scripts, while the oriental source contains three most frequently used languages: Chinese, Japanese and Korean.
- Language identification of documents within the oriental language category.

The results presented in this thesis were mainly obtained from analyzing and processing hundreds of documents (in 24 languages) that had been scanned and stored in the CENPARMI laboratory.

Language-category Classification between European and Oriental Documents

Sample documents in our database belong to either Roman, Cyrillic or oriental category. As mentioned before, Cyrillic documents do not possess the same upward concavity distribution as Roman documents. Therefore, they cannot be recognized by using the method of [Spi94] which was designed only to separate Roman and oriental documents. [LNB96] also did not deal with the Cyrillic documents.

For document classification between these two language groups, we investigated the features used in [Spi94], with an attempt to adjust and improve their features. However, we were unable to extend their results to process Cyrillic documents satisfactorily.

After further study and experimentation, we proposed several new features such as horizontal projection profile and height distribution of connected components. By using data analyses and processing, we are able to classify documents from the two language groups based on statistical features extracted from the documents.

Language Identification within Oriental Documents

Chinese, Japanese and Korean all belong to ideographic languages, i.e. they have a common structure of pictogram elements and each character can be viewed as a two dimensional image. As these three oriental languages have very large character sets, it is not feasible to apply token-based methods which have been used to differentiate Roman languages.

In [Spi94], “optical density” is the sole feature used to classify documents from the three oriental languages. As mentioned previously, this method is not good at dealing with documents in different fonts. It also requires an almost successful character segmentor as this optical density changes considerably from character to character. On the other hand, the method of [LNB96] had been used to separate only Chinese from Japanese, and it cannot properly differentiate between the three oriental languages. Again, this method depends considerably on the quality of character segmentation which cannot be guaranteed under existing research results. [SH98] proposed a good idea by using contour density to adjust the actual optical density, which will alleviate the problem with documents in multiple fonts. This method is still under study and more experiments are being carried out.

In our study and experimentation with documents of these three languages, we

have extracted and attempted the differentiation with a variety of features. In the process, we have selected three features that are effective in discriminating documents in Chinese, Japanese and Korean; these features are complex structure, Korean “circle” and long vertical stroke.

1.3 Organization of this Thesis

The remainder of the thesis is organized into the following chapters:

Chapter 2: Preprocessing

We describe the preprocessing methods which have been used.

Chapter 3: Differentiation between European and Oriental Scripts

The difference between European and oriental languages is discussed. We also describe our extracted features and how to apply them to differentiate these two classes.

Chapter 4: Differentiation of Chinese, Japanese and Korean

This chapter presents the visual characteristics of the three oriental languages. By analyzing the visual characteristics, we have proposed a method and conducted a series of experiments to classify these documents.

Chapter 5: Summary and Future Work

We summarize our work and contributions as well as indicate some future directions for future studies.

Chapter 2

Preprocessing

Our document images are obtained by scanning and binarization of text documents.

2.1 Noise Removal

Since the scanned images are usually not clean, we use the median filtering algorithm to remove noise. For each image, each input pixel is replaced by the median of the pixels contained in a window of 3X3 around it.

2.2 Segmentation

We must correctly separate text lines before analyzing their characteristics. The Run-Length Smoothing Algorithm (RLSA) [WWCS2] has been chosen to segment document images into individual text lines.

RLSA is the algorithm used to detect long horizontal and vertical white lines, hence locating the separated non-blank lines. Assume that white pixels are represented by binary 0's and black pixels by 1's. Within an arbitrary sequence of 0's and 1's, RLSA replaces 0's by 1's if the number of adjacent 0's is less than or equal to a certain predefined smoothing parameter S . For example, with $S = 3$, the sequence:

000110000101011000

is smoothed into the sequence:

111110000111111111

Horizontal and vertical one-dimensional smoothings are applied to the two dimensional input bitmap of the digitized document separately and two smoothed intermediate bitmaps are obtained. Then an "AND" logical operation is used to combine these two bitmaps. Due to the existence of some small gaps in certain text lines, an extra horizontal smoothing is performed to remove them. The choice of the parameters is not critical. The values of horizontal and vertical smoothing thresholds S_h and S_v should be set to a number of pixels covering the length of long words, whereas the threshold for removing small gaps S_g should be set to cover the width of a few character widths. Shown in A, B, C and D of Figure 2 are the horizontal, vertical smoothing bitmaps of an image with 989 pixels high, 1441 pixels wide, the bitmap of AND operation and the segmentation result after applying RLSA, respectively.

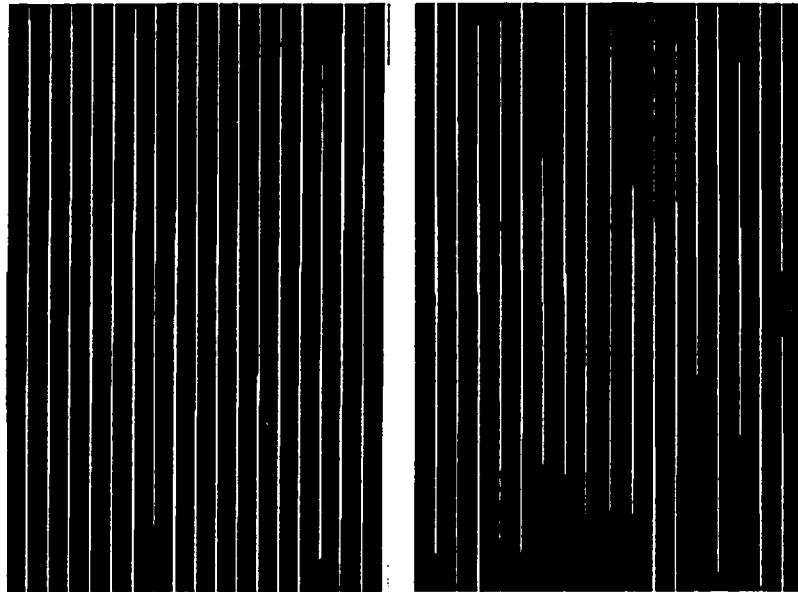


A: Horizontal Smoothing Bitmap($S_h = 211$)

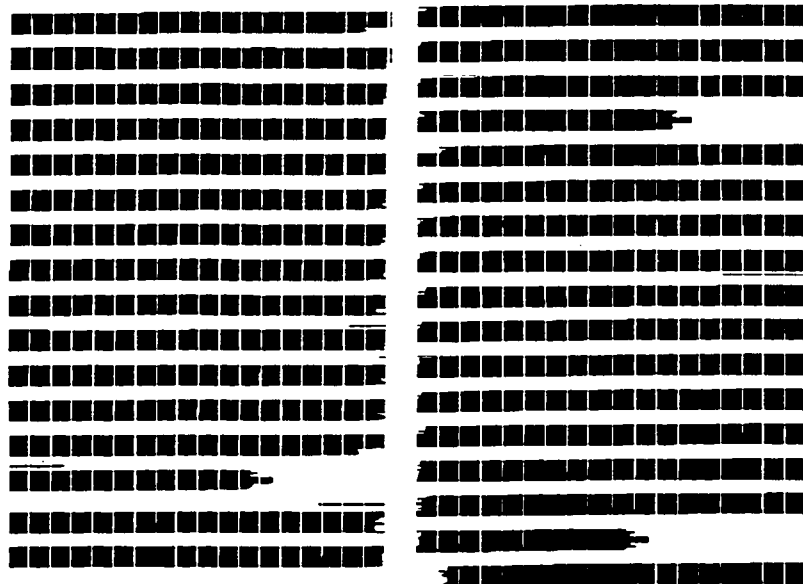
2.3 Deskewing

A text line is deskewed along its center of mass. Suppose a text line has height h and width w , and it contains N black pixels. Then its center of mass (\bar{x}, \bar{y}) is calculated as:

$$\bar{x} = \frac{1}{N} \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} x_i \cdot \bar{y} = \frac{1}{N} \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} y_j \quad (2)$$



B: Vertical Smoothing Bitmap($S_v = 241$)



C: AND Bitmap

文字是用來傳達意志，及思想的記錄，它也代表當代的文化。人人費盡心思來創造美麗的文字，使美麗的文字參加工商行業、宣傳、廣告等。爲了要產生更美麗的文字，因此有了文字設計這行專門學問。文字是供人閱讀的，歷史足以證明，人人歡迎美麗的文字，美麗的文字也活在每一個人的心中。觀賞用的文字有對聯、匾等掛在牆上，也是當作裝飾用；而書法足嚮引你走向這條路的指南，同時也保持藝術的崇高地位。書法與藝術的歷程一樣艱難，無論那一種藝術都應抱著崇高的理想，培養高深的技巧；練習書法也不例外，字體要反覆的練習以求進步。

目前在設計界佔著相當地位的圖案文字（也可以說藝術字），它的目的也是在創

視的。楷書、行書、草書等是表現「線」的流暢之美，線要寫得美，得靠筆底的應用，同時與速度也有關係；曲線和轉彎處還是慢慢地，沉著地寫較好。

要正確的表示點和聲，保持上下左右的平衡，並且使其有美的調和感，也是很重要的。根據「永字八法」來了解字體的性質，用筆畫的構成組織美麗的文字。文字有左右均衡與不均衡的分別，而同樣是一個文字，却因它的向勢與背勢，可變成不同的另外一種感情了。從各種筆畫中選出較有特徵的線來，強調這線的表現，看看有何不同，這種改變字體的嘗試也是很有趣的。寫字時要遵守筆順，從落筆開始以致完成，都需要注意運筆時的筆勢，這樣才能寫出有穩定感的字來。

文字沒有一定的基準，更沒有一定的形

D: Segmentation Result of RLSA($S_g = 21$)

Figure 2: Horizontal, Vertical Smoothing, AND Operation and Segmentation Result if and only if (x_i, y_j) is black.

The (p, q) order central moments $\mu_{p,q}$ is determined by the following:

$$\mu_{p,q} = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} (x_i - \bar{x})^p (y_j - \bar{y})^q \quad (3)$$

The orientation of a skewed text line is defined as the angle of axis of the least moment of inertia [Jai89], calculated by formula (4). By using this orientation angle θ , a skewed text line is aligned by the transformations of formula (5), where (x, y) are the coordinates before deskewing, and (α, β) are the coordinates after deskewing.

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \right) \quad (4)$$

$$\begin{aligned} \alpha &= x \cos \theta + y \sin \theta \\ \beta &= -x \sin \theta + y \cos \theta \end{aligned} \quad (5)$$

2.4 Line Concatenation

As with any statistical measure, reliability is a function of sample size. We try to accumulate data from those text lines containing enough information. However, sometimes all the text lines in a document may be very short. If this is the case, we need to

concatenate several lines into a long one. This is done along their baselines. Inaccuracies can occur if we simply concatenate lines along the bottom of bounding boxes. For example, when one line with descenders is to be concatenated with a line having no descenders, the concatenation based on the bottom of bounding boxes will not produce a straight line of text. Shown in (2) of Figure 3 is the concatenation of the two lines (shown in line (1) of this figure) along the bottom of their bounding boxes, while line (3) is the concatenation along their baselines. The difference between the results is clear.

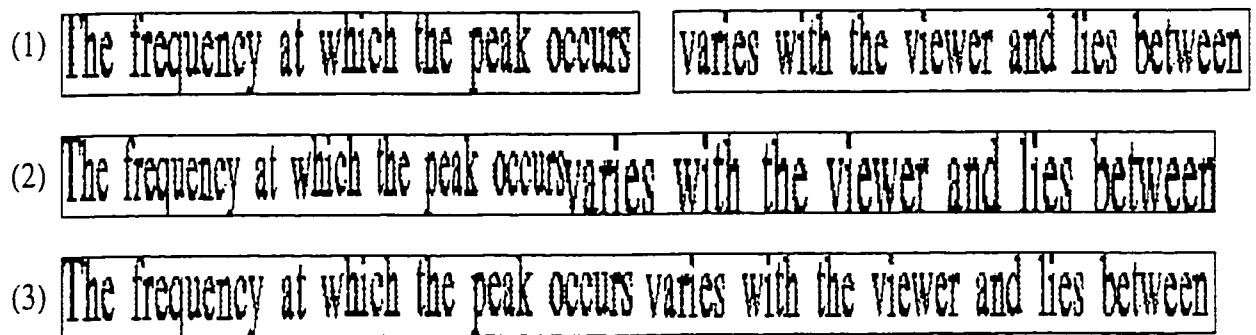


Figure 3: (1) Two Text Lines (2) Concatenation along the bottom of bounding box (3) Concatenation along the baseline

2.5 Size Normalization

All text lines are normalized to a given height. We use 64 pixels as the normalized height for all text lines as we want to express the height of the connected components as accurately as possible in order to facilitate distribution analysis when differentiating European and oriental languages. In order to retain the shapes of the original images, the width of each normalized image is calculated by $nor_width = w * 64/h$, where h and w represent the height and width of the original image, respectively.

Chapter 3

Differentiation between European and Oriental Scripts

After the image has been preprocessed by using the methods described in chapter 2, the next step of our work is to differentiate documents belonging to European languages from those of oriental languages.

Our work is based on the analysis of the differences between the language categories, and on experiments and results obtained by other researchers working in the same area.

Although it seems easy for a trained human being to recognize the differences between European and oriental documents, automatic classification of them by using computers is far from trivial. This is because human beings have superior intelligence including (but not limited to) deduction, analogy and judgement. They actually do not need to perform accurate calculations in differentiating the input documents. On the contrary, computers need to be given an accurate algorithm to “compute-and-compare” in order to render a decision. The problem of automatic language classification of documents has been non-trivial due to the fact that we have difficulty in translating our intelligence into a deterministic computer algorithm which consists of a series of steps of computations and if-then-else logic branches. In general, researchers try to explore the differences between these two language groups, make sure the differences (as represented by “features”) can be quantified (i.e. can be numerically represented), and are distinguishable after certain calculations performed by computers. Each method they proposed is based on a certain feature(or

features) they successfully extracted from the document images, and these are used to separate documents from different sources (language groups). The performance of different methods largely depends on the “quality” of the feature(s) they used, how well the features are extracted, and the decision-making process.

This chapter describes the work that we have done in studying the characteristics of documents in both European and oriental language groups, in searching for and extraction of features representing these characteristics, in experimental analysis/processing, and in decision strategy. Section 3.1 introduces the characteristics of the language groups under study, Section 3.2 covers the work that have been done based on the concavity feature, Section 3.3 describes the features that have been chosen to separate European documents from the oriental ones, Section 3.4 explains the combination of the features, while the experimental results and analyses are contained in Section 3.5.

3.1 Introduction

In our image database, sample documents are obtained from two sources: European and oriental. The European source is comprised of Roman and Cyrillic scripts. The writing systems of these two language categories are alphabetic, in which small sets of alphabetical characters are the primitive elements used to represent words, sentences, paragraphs and so on. On the contrary, the writing systems of the oriental category (Chinese, Japanese and Korean) languages are ideographical, an alternative to alphabets, in which each character can be viewed as a two dimensional image composed of a variable number of radicals [SMRW98].

There are no common alphabetic sets for the three oriental languages, instead each one has its unique writing style and basic patterns to build its characters. The Chinese characters are built from a combination or permutation of about 220 basic patterns [SMRW98]. Japanese is a mixture of Kanji (Chinese characters), Hiragana and Katakana. Korean script has evolved into pure syllabaries even though the structure of its characters (called Hangul in Korean) is similar to that of Chinese characters.

Compared with European languages, oriental ones have larger character sets and the characters are more complex.

3.2 Work Based on Concavity Feature

In [Spi94], the distribution of concavity points has been used to separate the oriental language category from the European one. According to [Spi94], when an image is scanned from its top to bottom, if two runs of black pixels appear on a single scan line and there is a run on the line below which spans the distance between these two runs, then an upward concavity is formed on the line. See Figure 4 as an example.

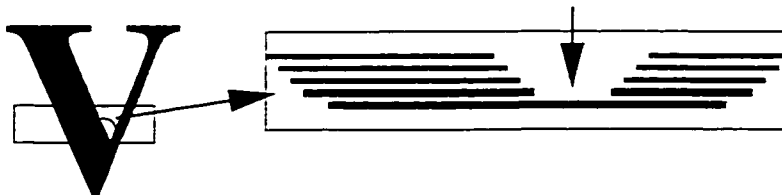


Figure 4: Upward Concavity of Letter V

Both Roman and Cyrillic scripts can be separated into three zones by several reference lines, as shown in Figure 5:

- Upper zone (ascender zone): the region between the top of text line and x -line.
- Middle zone (x zone): the region between the x -line and baseline.
- Lower zone (descender zone): the region between the baseline and the bottom of text line.

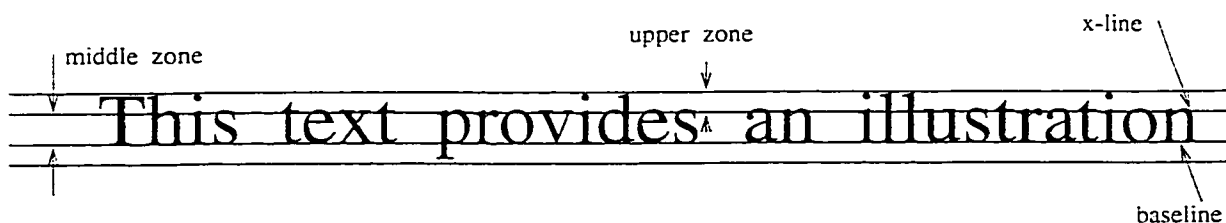


Figure 5: Reference lines separating European characters into three zones

Since Roman scripts are mostly composed of lower-case characters, their concavity points tend to cluster near the x -line and baseline, which is not the case for oriental scripts. Oriental characters contain more radicals located at different heights of the text line, yielding a more random distribution of concavity points. However, the concavity points of Cyrillic characters do not cluster only near x -line and baseline.

The different distribution of concavity points between Roman and Cyrillic scripts is due to the different shape of these two alphabets. For example, text line (2) of Fig. 6 shows a Bulgarian text line of Cyrillic script, in which the upward concavities cluster near the baseline and some positions between the baseline and the x -line. This distribution is quite different from that of the English text line shown in (1) of Fig. 6, where upward concavities only cluster near the x -line and the baseline. Shown in (3) of Fig. 6 is a Chinese text line, where the concavity points are more randomly distributed within the text line height.

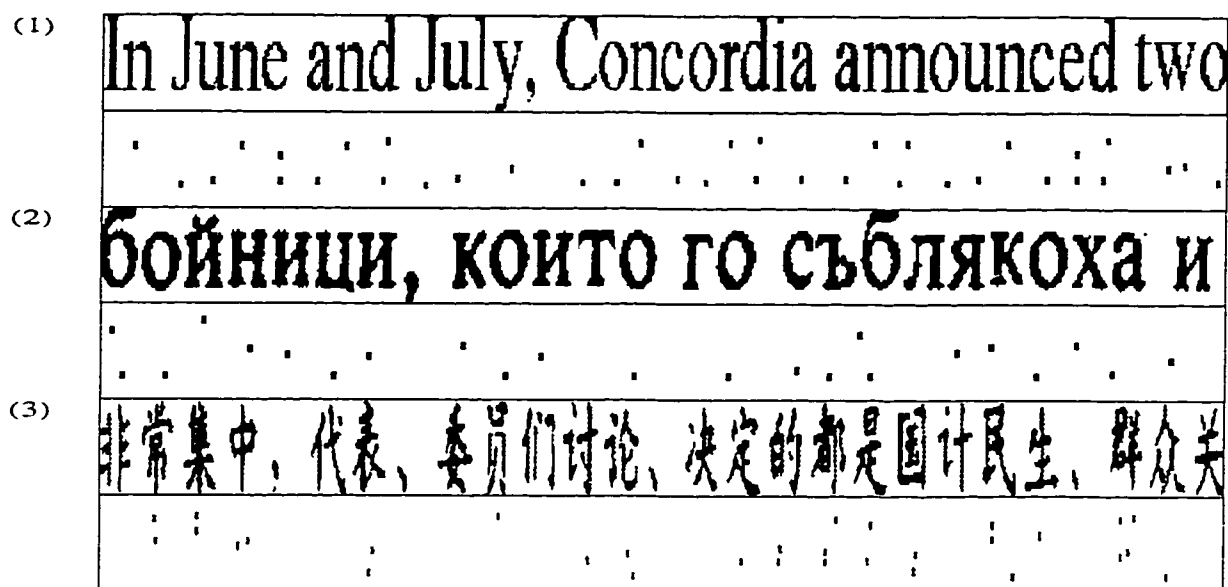


Figure 6: Upward concavity distribution of several text-lines

When the method of [Spi94] was applied to our document images, experimental results are not as good as those described in [Spi94]. We have tried to improve the results by enhancing concavity points qualification criteria as follows:

1. Black run length check: for a black run, its length should be greater than a certain threshold. By adding this condition, we can remove some short runs more likely to be caused by noise.
2. After locating the concavity points based on the definition of [Spi94], we further check that they are deep enough to be taken as “real” concavity points. This ensures that we find significant concavity points, and eliminates “cavity” points caused by uneven strokes.

After the above improvements, our method still cannot satisfactorily classify the two language categories with respect to the recognition rate and error rate.

Consequently, we proposed three other distinctive features and used them as the basis of our gross classification work. Experimental results show better classification performance than that of using concavity features. These three features will be described in Section 3.3.

It is worth mentioning that, after our work on gross classification has been completed, Lee et al. [LNB96] published their work in the same area. In this work, the distribution of concavity locations is one of the features used to distinguish between European and oriental language categories within their seven languages. Based on the observation of the different distribution of concavity locations between Roman and oriental scripts, the numbers of concavities located at different heights are determined for each text line, after which the locations are normalized by the text line height. Then the normalized distribution is considered, and the sum n_c of the numbers of concavities located in the two ranges 0.2 to 0.5 and 0.6 to 0.8 are determined. The position 0.2 means 20% below the top of the text line, and therefore 0.2–0.5 represents the x -line range while 0.6–0.8 stands for the baseline range. It is assumed that the regions outside these ranges are occupied by ascenders, diacritics and descenders of European scripts.

The decision based on this feature is made according to the following criterion. If n_c is greater than 0.8 of the total number of concavities, then the text line is classified as European. If n_c is between 0.6 and 0.8 of the total, then it is oriental. No decision is made otherwise.

We applied this method to perform the differentiation on our data sets. In a document, we randomly choose two relatively long text lines and the decision result is based on the majority voting obtained from the results of these two lines. We only process these longer lines because statistical results are more reliable when based on more information and data. Table 1 shows the results obtained from this method.

The results of Table 1 show that the error rate is low, but the rejection rate is relatively high. This agrees with the general result [LS97] that using an even number for majority vote gives a more reliable result, while it also produces a higher rejection rate when compared to the voting results obtained from using an odd number.

There are two particular problems with the above decision criteria:

Table 1: The results of language classification by concavity location

Language	# Samples	Recognition (%)	Error (%)	Reject(%)
European	272	59.19	5.88	34.93
Chinese	191	78.01	0.00	21.99
Japanese	94	47.87	5.32	46.81
Korean	164	36.59	0.61	62.80

- Those two ranges cannot handle lines containing only capital letters and/or lines without descenders. For each of these two cases, the baseline will move to a range greater than 0.8 of text line height.
- The above decision criteria cannot separate some Cyrillic scripts from oriental scripts as it does not consider the concavities lying between those two ranges, where concavities tend to accumulate for some Cyrillic scripts.

These problems are illustrated in Fig. 7 which has two text lines, one Swedish, the other Russian, and their concavity location histograms. The Swedish line does not contain any descenders, and therefore more than 30% of concavity points cluster near its baseline, which is located beyond 0.8 of its line height. In the Russian concavity histogram, 30% of the concavities accumulate in the range of 0.5–0.6. These two lines are misclassified as oriental.

Due to the above limitations, the position of concavity points was only used as one of the four features in the differentiation of two oriental languages (Chinese and Japanese) from the five Roman languages(English, French, German, Italian and Spanish) in [LNB96]. As our task is to handle a greater variety of languages, we need to explore more features in order to identify the language categories in our data sets.

3.3 Feature Description

In this section, we will discuss the features that have been used to separate European documents from oriental ones.

(1) natur erkänt att denna onda natur

(2) бойникам, которые сняли с него одежду, изранили его и ушли,

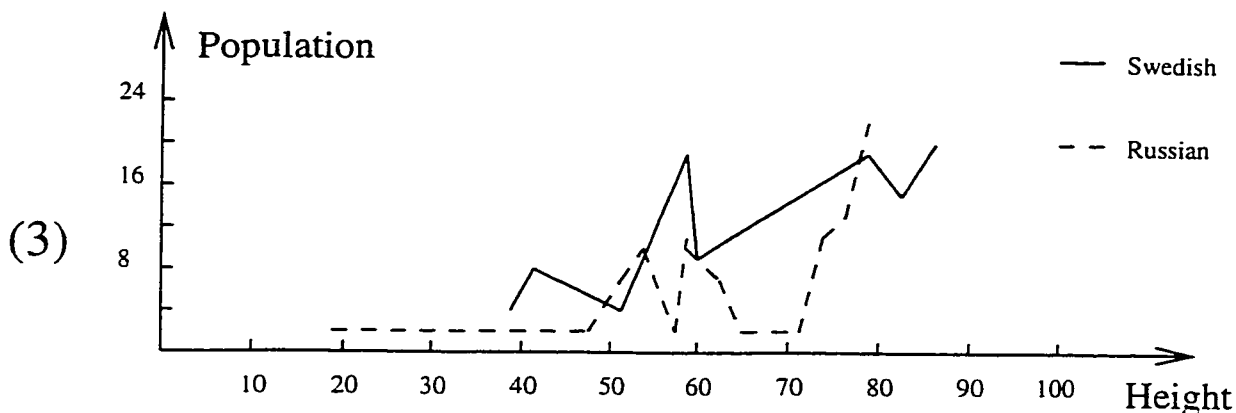


Figure 7: (1) A Swedish text line (2) A Russian text line (3) Concavity histograms

3.3.1 Horizontal Projection Profile

It is well known that one of the most distinctive and inherent characteristics of European alphabets is the existence of ascenders, descenders and diacritics. This characteristic has been used in the recognition of words in these languages (for example, [GS95]) and in the determination of character shapes to differentiate European languages, such as [SS94]. On the other hand, in oriental ideographical languages, each symbol is bounded by a square box and its height is limited within this box.

Assuming text lines are horizontally aligned and based on human observation of horizontal projection profiles partially illustrated in Fig. 8, we noticed that some differences between European and oriental languages are reflected in the following:

1. The location of local maxima at the lower and/or upper ends of the text line projection profile, and
2. Ratio of the area of white region bounded by the two most significant peaks to the area of projection profile.






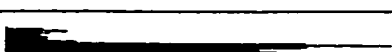
бойници, които го съблякоха и нараниха и отидоха си, като го	
下一步工作的部署是周密的、务实的。体现了解放思想、实事求是。	
る。この意味モデルは、筆者がデータ処理用に開発した意	
망에 대해 언급, 캐나	
Qu'est-ce qu'on fait quand on veut fabriquer un produit	
Papers should be at most 6 pages long. since this is the es-	

Figure 8: Several horizontal projection profiles

Local Maxima of European Profiles

European documents have local maxima (small peak) at the upper-end of their horizontal projection profiles. Below the small peak, a small basin area is present. For example, in the text line of Bulgarian script shown in the first line of Figure 8. we can find a small peak at the upper-end of the horizontal projection profile. For some European documents, there are lower-end small peaks as well, as shown in the profiles of French and English text lines in Figure 8.

Oriental documents, however, have no such small peaks. Refer to the profiles of Chinese, Japanese and Korean text lines in Figure 8.

The reason for European documents to have upper-end peaks in the profiles lies in the presence of ascenders, small over-hanging dots, accent signs, apostrophes, etc. The descenders of some European characters explain the existence of lower-end small peaks in some European profiles. For example, the “heavy” foot of “j” as well as that of “y” contribute to the lower-end peak in the horizontal projection profiles of European documents. Due to the fact that more European characters contribute to upper-end peak than those contribute to lower-end peak, the upper-end peak is more significant and can be detected more clearly.

Oriental documents have no local maxima at both ends in their projection profiles because oriental languages have large character sets and the uneven contribution to

different height position of horizontal projection profile by any character is mostly offset by other characters having different shapes and structures.

Based on this difference in the two language categories, we search for the upper-end and lower-end small peaks from the horizontal profile. The peak identification criteria are as follows:

- The upper-end and lower-end peaks can only be located in the upper-zone and lower-zone of the horizontal projection profile, respectively.
- The local maxima should be a relatively small value compared with the most significant peak in the whole profile because ascenders, descenders, diacritics, etc. of characters only form a small portion of a text line.
- A basin (minimum) must be found just below the peak. This basin must have a relative small value relative to that of the peak, and the distance between the position of peak and that of the basin must be of a certain value.

Here we give the algorithm of locating the upper-end peak in the horizontal projection profile. The procedure is analogous for the lower-end peak.

1. Locate an initial peak in the upper zone, i.e. in the position range of $([0, h/3 - 1])$ (h is the normalized text line height). To do this, we find an initial maxima at position i satisfying:

$$\begin{aligned} population[i] &\geq population[j], j < i \text{ and} \\ population[i] &\geq population[i + 1] \end{aligned}$$

2. Determine if the initial maxima is the candidate for the upper-end small peak.

If $population[i]$ is relatively big compared with the most significant peak in the whole profile (more than $1/3$ of the most significant peak) or if the width between position i and $h/3 - 1$ is too narrow (less than $1/10$ of h), then $population[i]$ is not a candidate and the algorithm terminates. Otherwise goto step 3.

3. Decide if this peak candidate is a real ascender peak as follows:

- The width between i and the immediate local minimum or the upper zone boundary ($h/3 - 1$) (if the boundary is reached before a local minimum is located) must be greater than $h/10$.

- The population of the immediate local minimum or the upper zone boundary should be less than $0.85 * population[i]$.

If such an ascender peak is found, then the algorithm terminates; otherwise the search is continued in the position range of $([i + 1, h/3 - 1])$ until the boundary is reached.

White-Black Ratio

In European languages, long black runs generally occur near the x-line and the baseline, resulting in two significant maxima at these positions. For the other positions along the text line, the horizontal projections have much lower values. In oriental scripts, the horizontal projections would not contain two such prominent maxima. It is known that oriental characters are denser than European characters and they have more *strokes*, especially more horizontal ones. For these reasons, there are more long black runs in an oriental horizontal projection profile than in a European one and they are also more evenly distributed from the top to the bottom of the text line, which makes the ratio of white area to black area smaller in oriental documents.

Consequently, if we consider the white space in the trapezoidal region bounded by the two maximum values (refer to Figure 9), then the ratio of white area to black area is larger in European than in oriental languages.

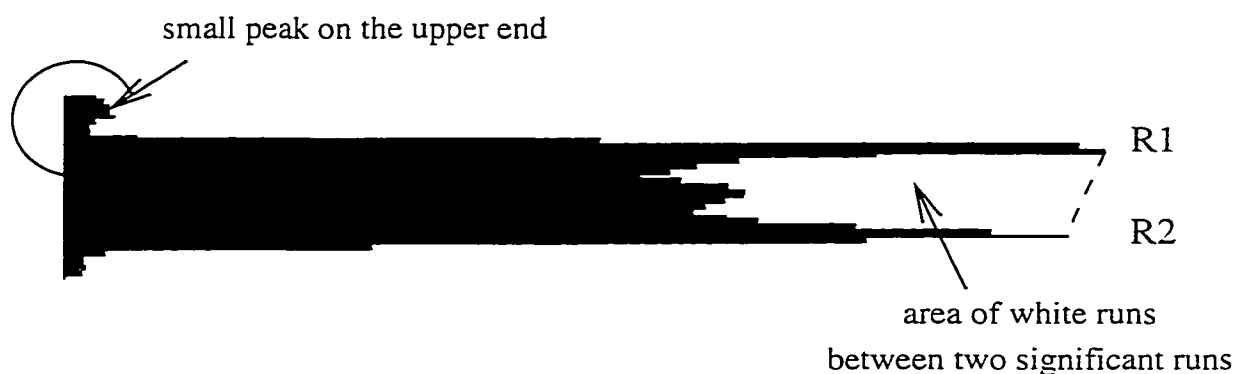


Figure 9: A horizontal projection profile of a Bulgarian script line

To calculate the black/white area ratio, the following method is employed:

1. Find the two maximum runs R_1 and R_2 on the upper half with position range $([0, h/2 - 1])$ and lower half $([h/2, h - 1])$ of the horizontal projection profile respectively, where h is the normalized text line height. In our case, h is 64.

- Determine the white area enclosed by R_1 , R_2 , and the line connecting R_1 and R_2 . This is illustrated in Figure 9. Assuming the trapezoidal region is bounded by $[h_{R1}, 0]$, $[h_{R1}, R1]$, $[h_{R2}, R2]$ and $[h_{R2}, 0]$, then the white area is:

$$S_w = \frac{1}{2} * (R1 + R2) * (h_{R2} - h_{R1}) - \sum_{i=h_{R1}}^{h_{R2}} R[i], \quad (6)$$

where h_{R1} , h_{R2} are the vertical locations of $R1$ and $R2$ respectively. $R[i]$ represents the length of horizontal black run at vertical position i .

- Determine the black area S_b of projection profile as:

$$S_b = \sum_{i=0}^{h-1} R[i] \quad (7)$$

The black area is the sum of all horizontal black runs in the range of text line height.

- Calculate the white/black area ratio R as:

$$R = \frac{S_w}{S_b} \quad (8)$$

The value of R is expected to be larger in European documents than in oriental ones.

3.3.2 Height Distribution of Connected Components

It is known that oriental characters are more complex than European ones. This kind of complexity can be represented by the number of connected components or the number of strokes in each character cell. For the 52 lower-case and upper-case letters of English alphabet, we know there should be only one connected component in each letter except letters “i” and “j”. By studying Roman and Cyrillic alphabets, we know that for the European scripts, most letters contain only one connected component and none of them should have more than three connected components, whereas the characters in oriental languages are composed of radicals (from 1 to more than 10, ([Sue92], [Eji94])), which can be located at different heights of the text-line, and these characters are more complex.

In our work, we do not use the number of connected components in each character to represent the difference between these two language categories, because this feature

relies on an almost accurate cell segmentor. Also, extracting strokes from these languages would be somewhat difficult. Instead, we use the height distribution of connected components to reflect the difference in complexity between oriental and European scripts.

For European scripts, the heights of the characters are mainly attributed to three factors:

1. Small accent signs.
2. Lower-case letters having neither ascenders nor descenders.
3. Upper-case letters and lower-case letters with ascenders or descenders.

Figure 10 shows a European text line and the generated connected components represented by bounding boxes.

30. Jedin človjek džěše z Jeruzalema do

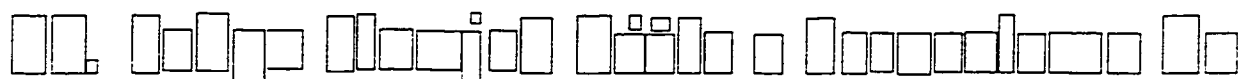


Figure 10: A European text line and its connected components representation

For oriental scripts, more than half of the characters consist of two or three connected components and these components can have many different heights, as illustrated by a Chinese text line and its connected components representation in Figure 11.

滴在酒内，一饮而尽。红军长征过大渡



Figure 11: A Chinese text line and its connected components representation

A comparison of Figure 10 and Figure 11 suggests that oriental scripts can be separated from European ones by the difference in the height distributions of their connected components. For a given text line, its height distribution of connected components is generated as follows:

1. All the eight-connected components are generated [RD84], and the coordinates and the dimensions of the bounding box for each component are determined.
2. The height distribution histogram of the components is constructed. Each value on the x -axis represents a height, which may vary from 1 pixel to the normalized image height. The y -axis shows the populations of components having each height.

Figures 12 and 13 show the height distribution histograms of three oriental text lines (Chinese, Japanese and Korean) and the height distributions of three European text lines (Polish, French and Russian). It can be seen that the height distributions of European languages show one or two prominent peaks representing factors 2 and 3 of the three factors stated previously, while those of oriental languages are more uniform and the outstanding peaks are located near the full text line height.

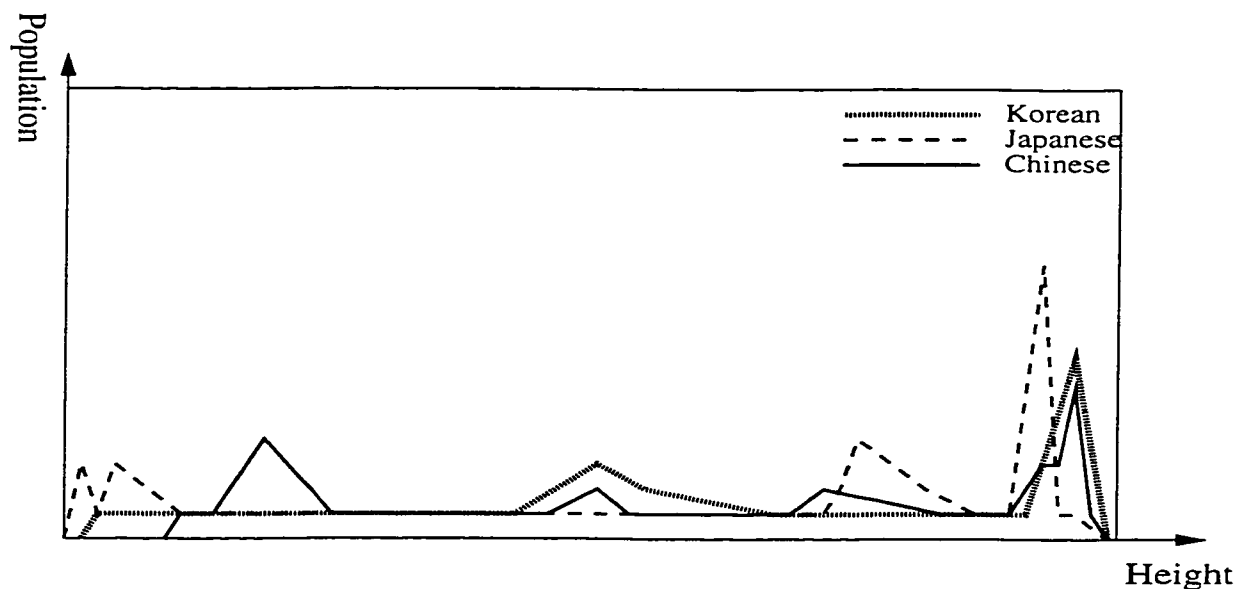


Figure 12: Height distribution of the components of oriental languages

We define a prominent peak as a value which is several times larger than the average with an absolute large value. The second condition is to ensure that we select a real prominent peak instead of a trivial one. (If we have a very small average value, several times of this value is still too small for it to be chosen as a prominent peak).

To search for prominent peaks in a height distribution histogram, the following procedures are adopted:

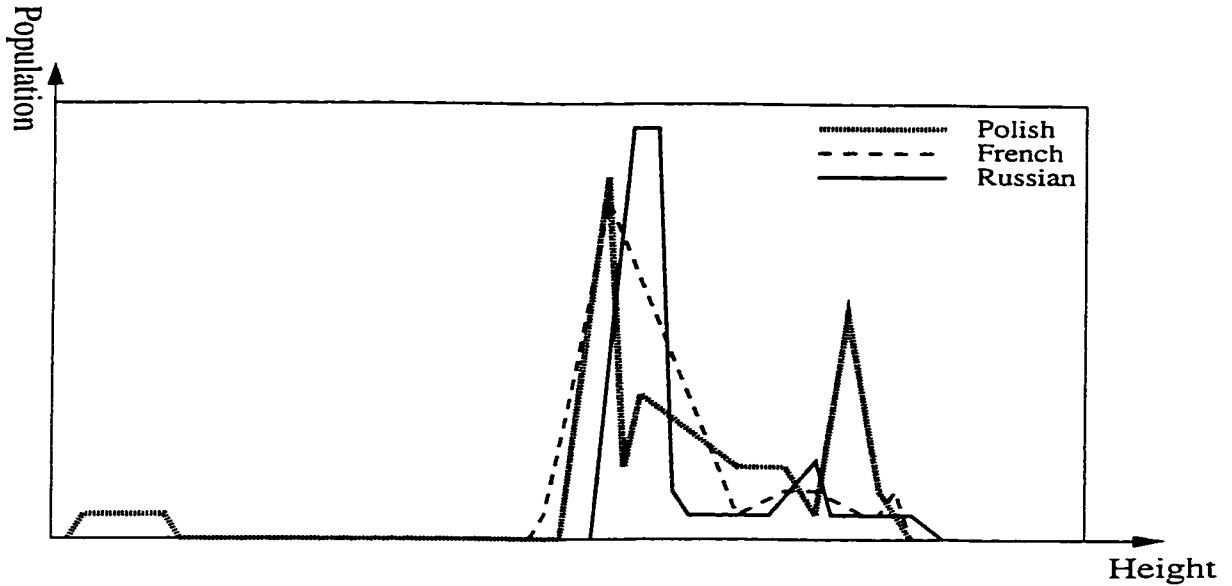


Figure 13: Height distribution of the components of European languages

- Height distribution histogram normalization. The x-axis has already been normalized to 64 pixels and we normalize the y-axis to 50. This value was chosen because we consider 50 components and the maximum possible height population is 50 if all the components have the same height.
- Calculate average values *avg* and *avg2*.
 1. Compute *avg*, the population mean.

Assume there are N different component heights in the height distribution histogram, then the *avg* is calculated as follows:

$$avg = \frac{1}{N} \sum_{i=1}^N population[i] \quad (9)$$

2. Compute *avg2*.

$$avg2 = \frac{1}{M} \sum_i population[i] \quad (10)$$

Where the summation is over all populations for which, $population[i] < MIN(2avg, th1+avg)$. In (10), *th1* is a threshold (*th1* is set to 4), and M is the total number of populations satisfying $population[i] < MIN(2avg, th1+avg)$. *avg2* is a partial average value of the population in the height distribution histogram. In calculating *avg2*, a few large peak populations

may be discarded, so that *avg2* can better represent the average value of a majority of populations in the histogram.

- Locate prominent peaks. A prominent peak $population[i]$ should satisfy the condition:

$$population[i] \geq MAX(th2 * avg2, avg + th3) \quad (11)$$

where $th2$, $th3$ are two thresholds obtained from the training data ($th2$ is 3 and $th3$ is 6).

- If prominent peaks cannot be located by the above method, then we take into consideration the two direct left and two direct right neighbors. If $population[i-1]$, $population[i-2]$ are the two direct left neighbors of $population[i]$ and $population[i+1]$, $population[i+2]$ are the two direct right neighbors, then we take $population[i]$ as a prominent peak if and only if:

$$\sum_{x=i-2}^{i+2} population[x] \geq MAX(th2 * avg2, avg + th3), \quad (12)$$

where the summation is taken over all x for which $population[x] > MIN(avg, avg2 + 2)$.

After locating prominent peaks from the height distribution of connected components, we classify this height distribution to European or oriental as follows:

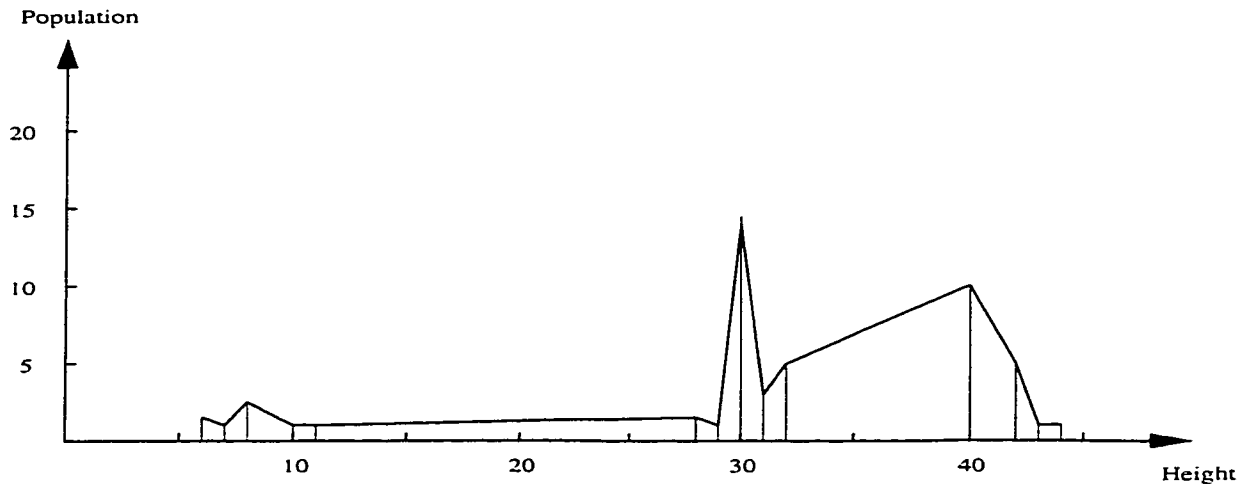
- Check the number of prominent peaks.
- Determine if the prominent peak is a qualified European peak which represents factors 2 and 3 of the European characters stated previously. The prominent peaks which appear below 1/4 or above 7/8 of normalized text line height are not taken into consideration because peaks located at less than 1/4 of text line height represent diacritics of European languages and those appearing at greater than 7/8 of text line height tend to belong to oriental characters.
- Determination based on the number of peaks:
 1. If more than two European peaks are found, then no decision is made for this height distribution;
 2. If one or two European peaks are found, then this distribution is classified as European;

3. Otherwise this distribution is classified as oriental.

We give an example to describe how the calculations are made to locate the prominent peak(s) and the classification result based on this information. Shown in Figure 14 is a European text line and its normalized components height distribution histogram.

Kiel elektigas la normlingvoj? La haveblaj historiaj atesto

(a): A European text line



(b): Components height distribution of (a)

Figure 14: A European text line and its normalized height distribution histogram

Based on the components height distribution, we compute *avg* and *avg2* according to equations (9), (10) and obtain two values: 3.57 and 2.17, where *avg2* is the average value of the remaining populations after discarding two peaks: 14 and 10. The next step is to search for prominent peak(s). Since $population[30] = 14$ and $population[40] = 10$, they satisfy equation (11), and are therefore considered as prominent peaks.

In the classification stage, our task is to determine if these prominent peaks are also European peaks and to make a decision based on the number of European peaks. Since these two peaks are located at heights 30 and 40, neither of them is located below $1/4$ or above $7/8$ of normalized text line height, hence they are considered to

be European peaks. This text line is classified as European because there are two European peaks in the height distribution histogram.

In summary, the component height distributions of European scripts possess at most two outstanding peaks, representing factors 2 and 3 stated previously. The component heights of oriental characters are distributed more evenly and a prominent peak occurs near the full text line height. This is because most oriental characters have a component occupying almost the full text line height.

3.3.3 Bounding Box Overlapping/Enclosing

Many oriental characters consist of two or more connected components and they are juxtaposed in various ways: right and left, top and bottom, one enclosed within another, etc. In contrast, all European characters have only up to three elements and one component is never contained within another. As shown in Figure 15, a sample Chinese character which means *nation* has three connected components, two of which are enclosed in a third one. This kind of situation does not occur for characters in European languages.



Figure 15: A Sample Chinese Character with Enclosed Structure

In order to locate the structure of one component enclosed within another in oriental languages, we use the following steps:

- Generate 8-connected components [RD84] and use bounding boxes to represent them.
- Determine if the “one component is enclosed within another” structure exists as follows:

Suppose two connected components $CC[i]$ and $CC[j]$, and suppose $ul(x_i, y_i)$,

$lr(x_i, y_i)$ and $ul(x_j, y_j)$, $lr(x_j, y_j)$ are the coordinates of the upper-left and lower-right corners of the bounding box of $CC[i]$, $CC[j]$. If

$$\begin{aligned} CC[j].ul.x_j &\leq CC[i].ul.x_i \leq CC[j].lr.x_j, \text{ and} \\ CC[j].ul.x_j &\leq CC[i].lr.x_i \leq CC[j].lr.x_j, \text{ and} \\ CC[j].ul.y_j &\leq CC[i].ul.y_i \leq CC[j].lr.y_j, \text{ and} \\ CC[j].ul.y_j &\leq CC[i].lr.y_i \leq CC[j].lr.y_j \end{aligned}$$

then $CC[i]$ is enclosed within $CC[j]$.

3.4 Combination of the Features

In previous sections, we have proposed three features to differentiate the two language scripts. Here, we will describe how these features are combined to produce our experimental results.

1. The horizontal projection profile is used to identify some European documents. If we can locate one and/or two small peaks and the white/black area ratio $\geq T_1$, then this document is classified as Roman. (T_1 is a threshold set at 0.38).
2. If no small peaks exist and ratio of white/black area $\leq T_2$, then the document is classified as oriental. (T_2 is a low threshold used to filter out some oriental documents).
3. If neither of the above is satisfied, then we consider the components height distribution. If this distribution shows an oriental tendency, then the document is classified as such. Otherwise,
4. If the components height distribution is Roman-like, then we consider the number e_{bb} of bounding boxes that are enclosed within another. If

$$e_{bb} \leq \frac{n_{cc}}{8} \tag{13}$$

the document is classified as Roman; otherwise it is considered to be oriental. (n_{cc} is the number of connected components considered).

3.5 Experimental Results

3.5.1 Datasets

Our images were scanned from books, newspapers, magazines and computer printouts. Both oriental and European documents are scanned at 300 dpi resolution.

In our database, there are 272 European document images printed in twenty-one different languages, such as English, French, German, Russian, Italian, Dutch, Portuguese, etc. For the oriental languages, we have 94 Japanese documents, 164 Korean documents and 191 Chinese documents.

3.5.2 Results

Tables 2, 3, 4 and 5 show the differentiation results on the basis of one 50-component, two 50 components, three 50 components and four 50 components, respectively. For the last three cases, decisions are made by majority vote of the 2 or 3 or 4 sets of 50 components. In our processing, we randomly select a relatively long text line and if necessary, several lines are concatenated to obtain 50 components. As the generation of one 50 components depends on random selection, we do not rely on the outcome of only one trial. So in the differentiation by using only one 50 components, we test the data set several times (in our case, we average the results of three trials) and average the results in order to reduce the element of chance. The average processing time for a document by using one 50 components is 17 seconds on a Sun Sparc 20 workstation.

Table 2: The results of language classification by using one 50-component

Language	Samples	Not processed	Recognition (%)	Error (%)	Reject(%)
European	262	0	95.32	4.68	0.00
Chinese	181	0	98.34	1.66	0.00
Japanese	84	0	99.21	0.79	0.00
Korean	154	0	97.62	2.38	0.00

Table 3: The results of language classification by using two 50-components

Language	Samples	Not processed	Recognition (%)	Error (%)	Reject(%)
European	262	1	95.40	0.38	4.22
Chinese	181	1	96.67	0.56	2.78
Japanese	84	0	100.00	0.00	0.00
Korean	154	0	96.10	0.00	3.90

Table 4: The results of language classification by using three 50-components

Language	Samples	Not processed	Recognition (%)	Error (%)	Reject(%)
European	262	2	98.08	1.92	0.00
Chinese	181	2	100.0	0.00	0.00
Japanese	84	0	100.0	0.00	0.00
Korean	154	5	100.0	0.00	0.00

Table 5: The results of language classification by using four 50-components

Language	Samples	Not processed	Recognition(%)	Error(%)	Reject(%)
European	262	5	99.22	0.00	0.78
Chinese	181	4	100.0	0.00	0.00
Japanese	84	0	100.0	0.00	0.00
Korean	154	10	100.0	0.00	0.00

3.5.3 Analysis of Results

At the present, we are able to differentiate the script of a document as being either European or Asian on the basis of four 50-components and obtain a high recognition rate while maintaining a relatively low rejection rate. From the results shown in Tables 2, 3, 4 and 5, we notice that the error rates are relatively higher when 1 and 3 units of 50 connected components are considered, while the rejection rates are higher when 2 and 4 units of 50-components are used to differentiate the two language groups. This voting results obtained also illustrate the theoretical findings published in [LS97]. When four 50-components are used, 100% reliability is achieved. It is also worth mentioning that an increasing number of documents cannot be processed when more units of 50-components are used for voting, because these documents do not contain enough components.

In our experiment, we find that when the quality of a European document image is low, either because many characters are broken or some characters are touching each other, then its component height distribution may cause the document to be misclassified as oriental script. For such a document, the number of bounding boxes enclosed in others may exceed the threshold set for European documents, thus causing the document to be classified as oriental script even if its components height distribution shows an European tendency. Also, we have difficulty identifying European documents written in certain fonts that are closer to handwriting than machine print. An example of this is shown in Figure 16.

Figures 17, 18 and 19 show examples of European text lines that are misclassified as oriental. The Swedish line of Figure 17 and the German line of Figure 18 are of poor quality as many characters are broken. The German line of Figure 19 is classified as oriental because many characters are touching, and also because the horizontal line beneath the first word causes the number of enclosed boxes to be large.

Oriental documents tend to be classified as European ones if more than 20% of their characters come from another source, such as European languages. In this case, a peak could be found which indicates the presence of European characters, causing the document to be misclassified as European. Figures 20, 21 and 22 show Chinese, Japanese and Korean text lines, respectively, all of which are classified as European lines because they contain a high percentage of European characters. Figure 23 shows a Chinese line which is identified as European because it contains a significant

proportion of digits.

Dans le but de clarifier et préciser les rôles et fonctions de l'organisateur (trice)

Figure 16: A French Line Misclassified as Oriental

vaksam. I Hou drog sig bort från honom, så försikti

Figure 17: A Swedish Line Misclassified as Oriental

d) Ambrosianus D (Sign. G 82 parte superiore) S H attc

Figure 18: A German Line Misclassified as Oriental

Außenfinanzierung: Das Kapital kommt aus Kapitaleinlagen oder Kreditgewährungen.

Figure 19: A German Line Misclassified as Oriental

在生产制造领域中,DEC 公司宣布了三种 MANMAN 系统的汉字版本,MANMAN/MFG,MAN-了标准存取方法。

Figure 20: A Chinese Line Misclassified as European

ロー図 (communication flow diagram) は, イベントトレ本論文では, 最初に,

Figure 21: A Japanese Line Misclassified as European

sorry I misplaced your business card.(아임 쓰오리 아싼다. 글자 그대로 풀이하면 잘못(mis) 든다(pl

Figure 22: A Korean Line Misclassified as European

名字”,于 1961 年创设了人名用汉字(92 个),1971 年又决定允许使用常用汉字 1945 个和人名用汉

Figure 23: A Chinese Line Misclassified as European

Chapter 4

Identification of Chinese, Japanese and Korean

By using the features and classification method described in Chapter 3, our document images have been separated into two groups: European and oriental. Our next step is to identify each of the three languages in the oriental group, i.e.: Chinese, Japanese, and Korean. The writing systems of these languages have a common structure of pictogram elements [SMRW98], and they are further inter-related through usage of a common set of characters.

One reason which makes the identification of these three oriental languages relatively difficult is that they have very large character sets which are also increasing in size. This is not the case for the European languages, in which the character sets are small and closed sets. Due to this reason, many methods which are effective for the differentiation of European languages cannot be adopted and applied to the classification of these three oriental languages.

In this chapter, we will introduce the identification of these three oriental languages based on an analysis of the visual characteristics of their characters.

4.1 Characteristics of Chinese, Japanese and Korean Characters

The three oriental scripts can all be called ideographical or pictographical scripts while each of them possesses its own characteristics which can be used to distinguish

it from the others.

Chinese characters are unique because of their special system of construction, their long history (more than 4,500 years) and their large number. One unique feature of Chinese characters is their appearance. Each character is formed by a definite number of strokes, usually less than 12 in number, but some characters have more than thirty strokes [Wan88]. Regardless of their number, all strokes should fit into a square box in an appropriate way. The result is that each character remains within its own box, and characters do not overlap. Because of this, Chinese characters are also known as “square characters”. Other unique features of Chinese characters are the large vocabulary used nowadays, and also that the Chinese vocabulary is open-ended and continues to grow, although at a slow pace. According to the study of *Chinese Character Analysis Group of Taiwan*, there are more than 74,000 Chinese characters known today, each represented by a unique graphic picture. However, less than 5,000 common characters cover 99% of daily usage, and the most 2,000 common characters constitute 97% of usage [SMRW98], [HH].

Chinese characters were exported from China to Japan more than a thousand years ago [SMRW98] and since then, Chinese characters, called Kanji, were used as a standard in Japan. However, among the large number of Chinese characters, at most 4,000 are used in Japan. Besides Kanji, two sets of Japanese Kanas (Hiragana and Katakana) are also used in Japanese. Consequently, Japanese is actually a mixture of Kanji (for, most nouns, verbs, and adjectives), Hiragana and Katakana which are phonetic symbols invented by the Japanese to represent particles and other grammatical parts which have no exact equivalents in Chinese. Katakana is an angular script which is most often used for borrowed words and emphasis, while Hiragana is a cursive script with sounds correspond to those in Katakana. Katakana and Hiragana sets contain 86 and 83 characters, respectively [Lau93].

According to [SMRW98], there was no distinct Korean alphabet until the 15th century and Koreans used Chinese ideographs to express their languages in writing, sometimes to represent the ideographs’ original meaning and sometimes to simply express sounds. However, the sounds of the Korean languages differ from those of Chinese, and the learning of writing complex Chinese characters was a difficulty for the common people. In order to solve this problem, King Sejong the Great appointed a group of scholars to invent a new simple method of writing down spoken Korean. As a result, the Korean script, called Hangeul was devised. Hangeul is a highly phonic

writing system [Chu91] in the following aspects:

- Hangul was invented based on the spoken Korean language.
- The shapes of Hangul alphabet symbols were created to imitate human speech sounds.
- The articulatory principle of syllables and words is explicit from the texture notation.

There are 11,000 possible Korean characters, only 2,300 of which are commonly used. The Korean language can be viewed as a hybrid of the Roman alphabetical language and the ideographical Chinese script. The lexical structure of Korean is analogous to that of Roman languages because of the hierarchical formation of the alphabet, phonemes, morphemes, and words. However the graphical form of Hangul characters is similar to that of Chinese, as each character is a two-dimensional arrangements of letters within a square box. For these reasons, Korean script is a phonography with an ideographical flavor.

4.2 Work Based on Optical Density

By studying the characteristics of Chinese, Japanese and Korean scripts, we know that Chinese characters are the most complex, and searching for a proper feature to represent this kind of complexity becomes one of our tasks in the language discrimination process.

From the previous section, we know that Chinese texts are composed of predominantly dense Chinese characters, Japanese characters are a mixture of relatively light characters(including Katakana and Hiragana) with relatively dense characters (Kanji). Based on this observation, Spitz [Spi94] proposed the differentiation of the 3 oriental languages by using an optical density feature as described in Section 1.1.2. This method has two main drawbacks:

- Optical density is decided not only by the language to which the character belongs, but also by the font and printing style. Different fonts in the same language may have different optical densities.

- Optical density relies on a successful cell segmentor as the density varies from character to character.

Due to the above reasons, Lee et al. [LNB96] modified the definition of optical density (see Chapter 1 for a detailed description) to take into consideration both pure optical density and the difference introduced by font design, and used the new definition to separate Chinese from Japanese.

We have applied the method of [LNB96] to our Chinese and Japanese training samples and obtained the results shown in Figure 24, from which it can be seen that Japanese and Chinese training samples can be separated linearly. Unfortunately, this method is only effective for the separation of Chinese and Japanese texts. When we add the Korean training samples, we obtain the results of Figure 25, which shows that Korean cannot be distinguished from Chinese and Japanese.

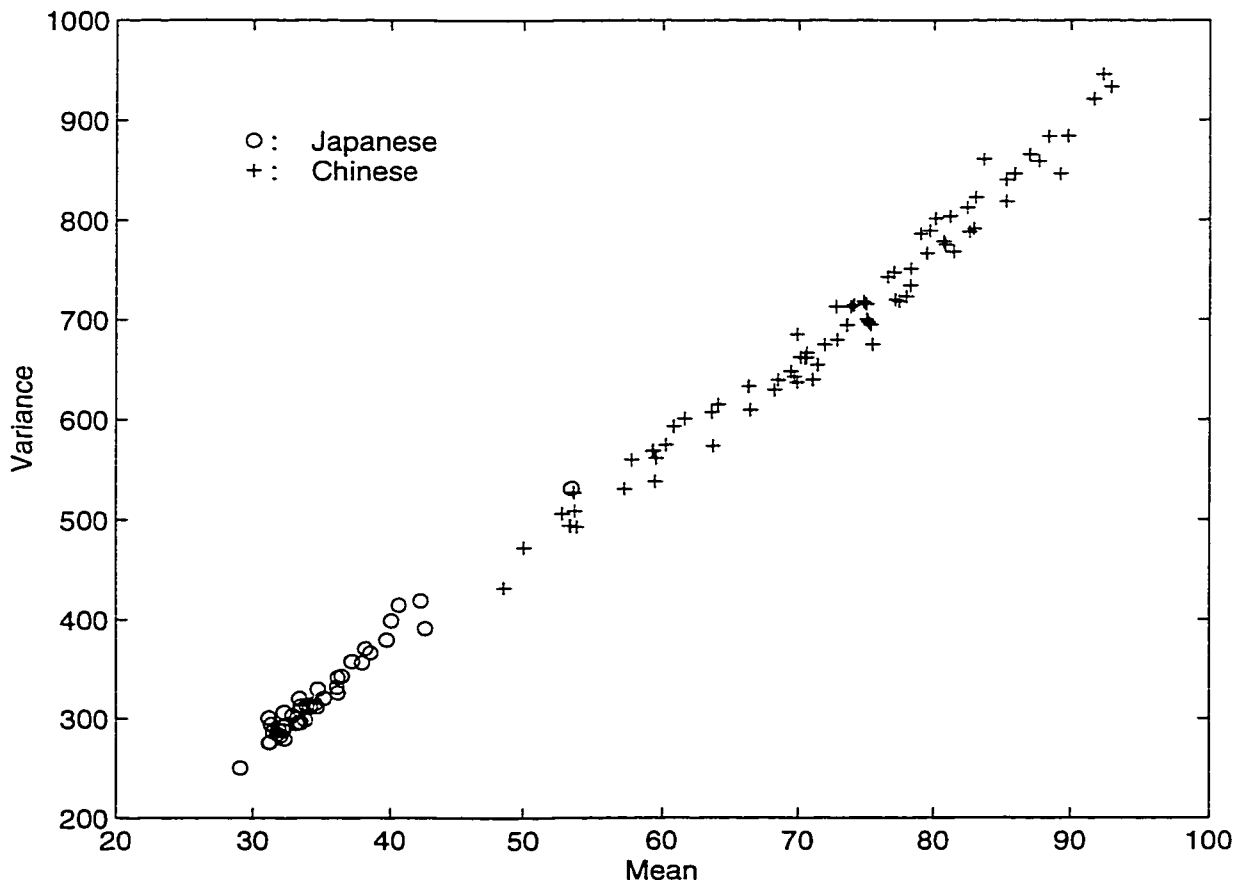


Figure 24: Method of Lee et al. when applied to Chinese and Japanese training samples

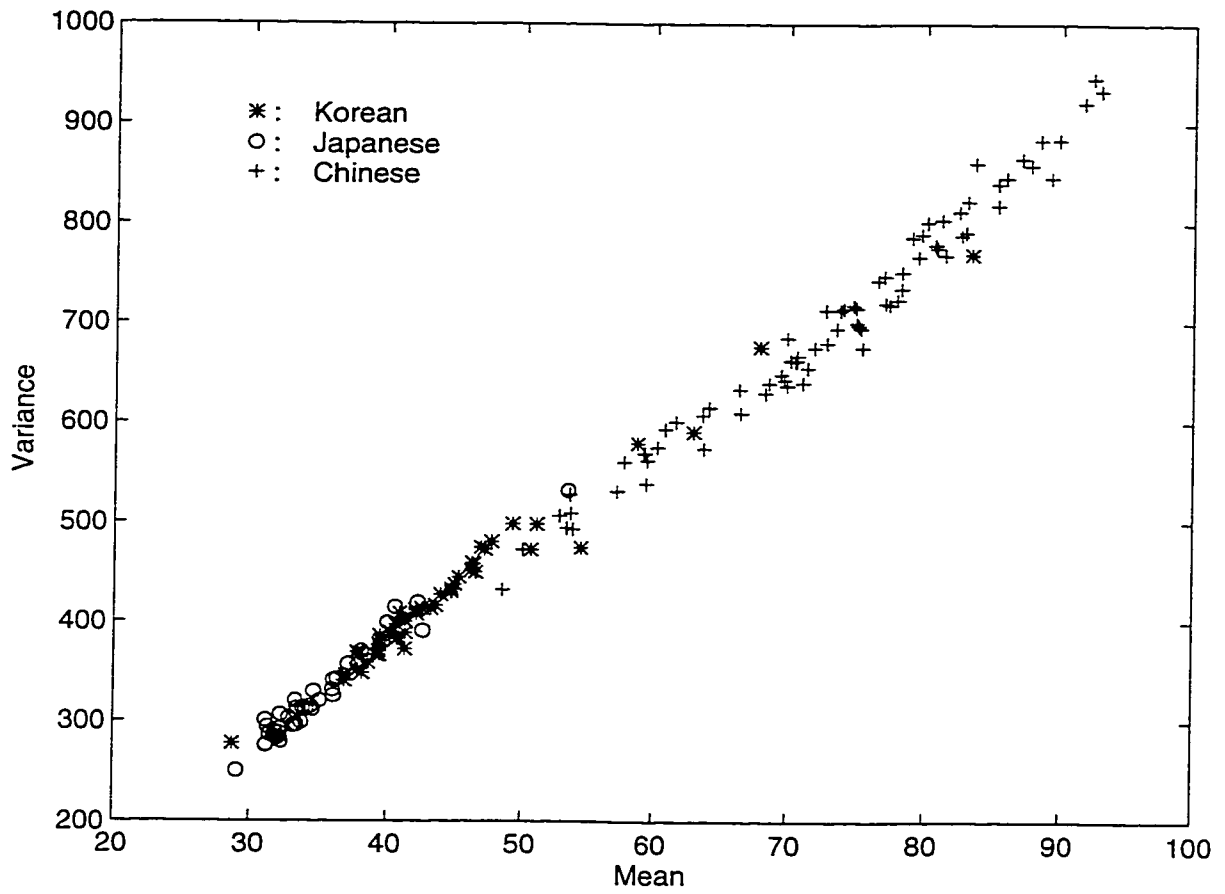


Figure 25: When applied to Chinese, Japanese and Korean training samples

4.3 Feature Extraction

In this section, we will propose some new features to identify the three oriental languages. In our feature extraction process, we obtain the character cell by computing the vertical projection profile of the text lines. Assuming the text lines are oriented horizontally, each character cell can be separated by all zero entries or white gaps as they are called, as illustrated in Figure 26 for a Japanese text line.



Figure 26: Character cells of a Japanese text line

4.3.1 Complexity of Structure

This feature emphasizes the complexity of Kanji characters in Chinese and Japanese languages and differentiates them from the simpler Korean, Hiragana and Katagana characters. A character cell is said to have a complex structure if it has at least one loop containing other components, or this loop contains more than one inner contour. Figure 27 presents several examples of Chinese characters which have a “complex structure”. The first and second characters, meaning “*blood*” and “*self*” respectively, contain three small rectangles within their outer contours. For simplicity, all squares, rectangles, circles, and ellipses will be treated as loops from here onwards. The third character, meaning “*return*”, has a loop enclosing another one: the small rectangle. To extract the loop features, we apply the contour tracing algorithm [Str93], which



Figure 27: Examples of complex structure

determines the connected components and stores them in a representation which reflects their topological relationships. The relationships captured include the order of occurrence of the leftmost pixel of the uppermost part of each contour as the image is scanned row by row, and the nesting of contours within others. Any outer contours

contained within an inner contour are linked horizontally to its right, while any inner contours contained within an outer contour are linked vertically beneath it. The complexity of a character can be obtained by checking its contours as follows:

- At least one of the character’s outer contours has more than one element linked below it, meaning this outer contour contains more than one inner contour, e.g., the first and second characters in Figure 27;
- At least one of the character’s inner contours has at least one element linked to its right, meaning this inner contour has at least one outer contour, e.g., the third character in Figure 27.

If one of the above conditions is satisfied, we say that we have found a character possessing a “complex structure”. Shown in Figure 28 is an example of a Chinese text line, in which all the characters with complex structure are marked by bounding boxes. The fact that Kanji characters contain more *strokes* juxtaposed in various ways results in the frequent occurrence of characters with complex structures. After experimentation, we determine that the frequency of the complex structure in Kanji characters is greater than that in Hangul characters, which in turn is greater than that in Hiragana and Katagana characters.



Figure 28: Characters with Complex structure in a Chinese Text Line

4.3.2 Korean “circles” and “ellipses”

“Circles” or “ellipses” are among the primitive *strokes* used in the Korean script. According to [KK96b], the frequency of their usage is 19.47%. The fact that many Korean characters contain “circles” or “ellipses” was noted in the very early stage of our oriental script differentiation process, but extracting a proper feature to represent this information had taken thought and effort.

Digitized “circles” and “ellipses” are not the perfect circles or ellipses as in geometry, because they are actually very similar to digitized “squares”, “rectangles” and other loops. The fact that Kanji characters contain many “squares”, “rectangles” and

other loops has necessitated a search for methods to represent the difference between Korean “circles”, “ellipses” and other loops based on more than their shapes alone.

Size Filtering

Loops which are too short tend to be small punctuations or small portions of some Kanji, Katakana and Hiragana characters (see line 1 of Figure 29 for some examples). At the same time, loops with heights greater than half of the text line height tend to be part of the most frequently occurring Hiragana characters (see character 2 of line 2) and some Kanji characters (see character 1 of line 2).

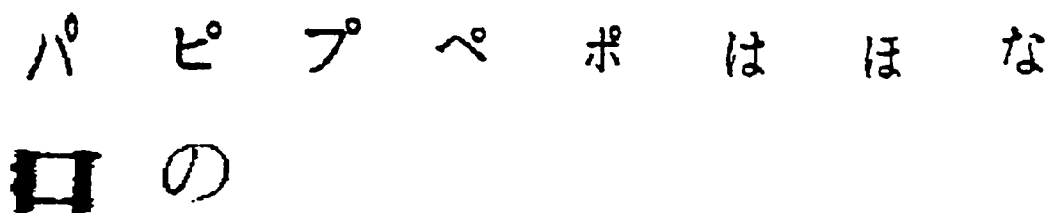


Figure 29: Examples of characters containing short and tall loops

Consequently, we choose the isolated loops occupying less than half of the text line height and more than a minimum height tolerance as eligible candidates for further processing. This first step of filtering can quickly reject those loops which are either too short or too tall. The remaining loops are then considered below.

Location Filtering

Korean characters are composed of 24 simple graphemes and 27 complex graphemes consisting of two or three simpler graphemes. A grapheme is either a vowel or a consonant [KK96a]. Ten of the simple graphemes are vowels and the rest are consonants. Korean characters can have six structures according to the grapheme combination as shown in Figure 30, where VV denotes a vertical vowel, HV a horizontal vowel, C1 the first consonant and C2 the last consonant. For every Korean character, there must be one first consonant and at least one vowel [KK96b]. The first consonant should be on the left of the vertical vowel (see Types 1, 2, 5 and 6 in Figure 30) and on top of the horizontal vowel, if it exists (see Types 3, 4, 5 and 6). The optional last consonant is located below the first consonant and the vowel (see Types 2, 4 and 6).

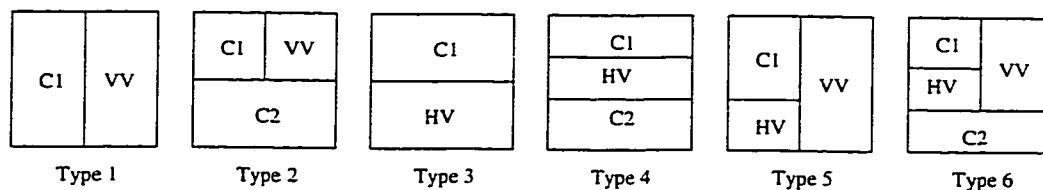


Figure 30: Six structures of Korean characters

Korean “circles” and “ellipses” occur only in Korean consonants. If this consonant appears as the first consonant in a character, then from Figure 30, we know it is located at either the very left (Types 1, 2, 5 and 6) or very top (Types 2, 3, 4, 5, 6) positions. If it appears as the optional last consonant, then it will be located at the very bottom (Types 2, 4 and 6) position of its character within the bounding box. From this, we can conclude that many Korean “circles” and “ellipses” appear at the very top, bottom or left of its character cell. On the other hand, Kanji “squares” and “rectangles” do not appear in specific positions, but are randomly located in its character cell.

In our method, we only analyze those isolated loops located at the very top, bottom or left of its character cell. If any external contour of a character contains exactly one internal contour and this internal contour does not contain any other contours, then the character has an isolated loop. During the character segmentation phase (in which each character cell is obtained from its vertical projection profile, assuming the text line is horizontal), the coordinates of the top-left point, width and height of the character’s bounding box are determined. When we analyze the position of a loop, we consider the bounding box of its inner contour instead of its outer contour. Since a loop’s outer contour may be connected to other part(s) of the character, it would not be useful if we use this bounding box to determine the position of the loop. This can be seen in Figure 31, which illustrates bounding boxes of external contours of loops. For this reason, we consider the inner contours of loops, and we need to consider a loop’s absolute position as well as its position relative to the character cell containing it.

Let X and Y represent the vertical and horizontal axes, respectively, where X increases from top to bottom and Y goes from left to right. Let (x_0, y_0) and (x_1, y_1) denote the top-left and bottom-right points of a character cell’s bounding box, respectively. Suppose (x_l, y_l) and (x_b, y_b) are the top-left and bottom-right points of the bounding box of the loop’s inner contour. Then we calculate three ratios to determine



Figure 31: Loop’s external contour containing other parts

the position of the loop relative to the character cell using the bounding box of the inner contour of the loop.

- $(y_1 - y_l)/(y_l - y_0)$ is used to determine the closeness of the left of the loop to that of the character. If this ratio is at least r_1 ($r_1 = 7$), then we consider this loop to be close to the left of its character cell.
- $(x_1 - x_l)/(x_l - x_0)$ is used to determine the closeness of the top of the loop to that of the character. If this ratio is at least r_2 ($r_2 = 4$), then we consider this loop to be close to the top of its character cell.
- $(x_b - x_0)/(x_1 - x_b)$ is used to determine the closeness of the bottom of the loop to that of the character. If this ratio is at least r_3 ($r_3 = 10$), then we consider this loop to be close to the bottom of its character cell.

By using the relative positions of the loop, the impact of unpredictable stroke thickness is alleviated.

After this position filtering, loops which are not located at the above specific positions would be filtered out. Figure 32 shows a Japanese text line, in which the loops belonging to characters 1 and 3 are filtered out after size filtering, and loops belonging to characters 2 and 4 are rejected by position filtering. After size and

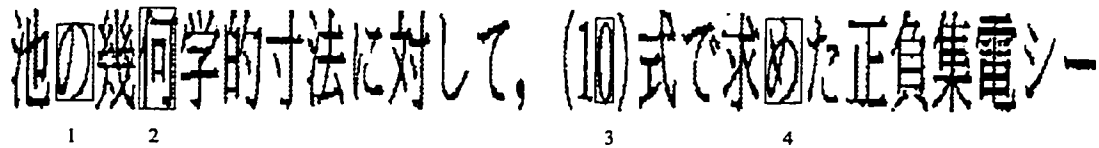


Figure 32: A Japanese text line

position filtering, a large number of isolated loops would have been rejected, but there may still remain some loops which are neither Korean “circles” nor “ellipses”. Figure 33 gives examples of such loops, and characters containing them are indicated by bounding boxes.

을 소개한다. 일반적인 작업 방법, 다양한 정보와 함

Figure 33: A Korean text line

Turning angle of corner points

After the previous two steps based on character size and position have been applied to identify possible Korean “circles” / “ellipses”, the detection strategy is based on character shape. Theoretically, a square or a rectangle is formed by four straight lines, has four corner points and the turning angle of each corner point should be or very close to a right angle. Geometrical circles are constructed by smooth curves and they should have few straight lines and corner points. Hence from the number of corner points or straight lines, we should be able to separate squares and rectangles from circles.

Unfortunately, digitized images often do not possess perfect geometric shapes. The numbers of corner points of Korean “circles” / “ellipses” are not fixed, and the number of straight lines constituting “squares” / “rectangles” may also vary. Because of these reasons, it is impossible to differentiate “circles” / “ellipses” from “squares” / “rectangles” simply by the number of corner points or straight lines. Therefore, we consider the turning angle at each corner point.

We use the method proposed in [Str93] to detect the contour corner points. In the first step, points of high curvature are located. As the algorithm tends to find a cluster of points in the vicinity of a corner instead of just one point, the second step is to merge these clusters. In this step, the contour is traced and corner points that are less than a small threshold distance t_c from each other are grouped into the same cluster. Each cluster is then reduced to those corners in the cluster for which the absolute N-code [GN70] is greater than a threshold:

$$|c_i^N| = |c_i^{t_c}| \geq 2t_c \quad (14)$$

where the N-code is one way of using differential code to locate corners. It computes a weighted sum of differential chain codes in a sequence of pixels of length $2N - 1$ centered at the i^{th} pixel in the contour:

$$c_i^N = Nc_i + \sum_{k=1}^{N-1} (N-k)(c_{i-k} + c_{i+k}) \quad (15)$$

where c_i is the differential chain-code at the i^{th} pixel.

After detecting contour corner points, we inspect their turning angles. We assume a larger turning angle indicates smooth turning at the corner while a smaller angle represents a relatively sharp turning.

For a contour corner point A, we search for points F and B along the chain in order to define the turning angle of A. Neither F nor B should be A's adjacent corner points because using A's adjacent corner points would produce information that is too localized to allow decisions to be made on the turning angle of A. F and B are on the left and right sides of A, respectively, if clockwise contour tracing direction is followed. The selection criteria for points F and B are based on (a) the length of chain codes, and (b) the positions of A's adjacent contour corner points. We define *skip*:

$$skip = MAX(3, chainlength/24)$$

as the number of points needed to be skipped in order to get B and F, also B and F should not be beyond A's adjacent corner points. After locating B and F, the turning angle θ of A is computed as:

$$\theta = \arccos((d1^2 + d2^2 - d3^2)/(2 * d1 * d2)) \quad (16)$$

assuming $d1$, $d2$ and $d3$ are the length of lines AF, AB and BF, respectively. Through experimentation, a loop is classified as a Korean "circle" if it satisfies the following two conditions:

- Turning angle of each contour corner point is greater than or equal to *degree1* and
- The average turning angle of all corner points is greater than or equal to *degree2*,

where *degree1*, *degree2* are set to 100 and 120 in degrees, respectively. Figure 34 illustrates a Korean character (marked by a box) containing a rectangle which is distinguished by sharp turning angles of its corner points.

This method can efficiently differentiate Korean "circles" from rectangles, squares, triangles and some other loops. Unfortunately, this method cannot separate Korean "ellipses" from rectangles/squares and some other loops when the turning angles of corner points on Korean "ellipses" do not follow the above two rules. The larger the


일 구 러 장 적  제 주 과 동

Figure 34: Rectangle contained in a Korean text line

curvature of ellipses, the more difficult it is to separate them from other loops by this method. Therefore this method may not be effective in certain Korean fonts, where “ellipses” are used instead of “circles”.

Determination of Korean “circles”

In our method, a Korean “circle” is determined by the following rules:

1. For an isolated loop, if its height is less than half of its text line height and greater than the minimum height tolerance, then go to step 2;
2. If it is located at the very top, or very bottom or very left of its character cell, then go to step 3 or 4;
3. If the isolated loop contains no corner point, then it is classified as a Korean “circle”;
4. If the isolated loop contains corner point(s) and all corner points have large turning angles then it is classified as a Korean “circle”.

4.3.3 Korean Vertical Strokes

This feature is designed to target a group of frequently used Korean characters containing certain unique connected components. The decision to choose this distinguishing feature was based on experiments. However, from the following paragraph, we can conceptually understand why it helps us in differentiating scripts of the three oriental languages.

Korean Script

According to observation and analysis, a significant number of characters in Korean script contain isolated vertical strokes, and a large proportion of such strokes are almost as tall as the Korean characters themselves.

이	다	하	는	을	지	에	가	의	고
기	어	리	들	아	나	은	자	사	도
한	시	라	었	를	해	보	있	그	대
인	서	부	게	정	로	것	수	여	으
일	구	러	장	적	면	제	주	과	동
만	생	우	되	했	상	람	전	야	문
려	마	무	음	거	각	요	내	소	원
신	까	성	국	화	물	위	치	오	말
스	모	였	르	용	계	조	선	비	와
았	교	희	간	경	관	학	니	방	산

Figure 35: 100 Most Frequently Used Korean Characters

Figure 35 shows the 100 most frequently used characters in modern Korean script. As we can see, about 12% of the characters contain a completely isolated long vertical stroke, normally on the righthand side of the characters. Also, another 12% of characters contain similar vertical strokes with one or two horizontal “bars”. These two kinds of isolated vertical strokes add up to more than 24% of the 100 most frequently used characters. We believe this is very distinctive and should be very useful in identifying Korean script.

Chinese Script

It is well known that the Chinese character set is extremely large. About four thousand characters are considered to be most frequently used and are indispensable for composing ordinary Chinese documents. In these most frequently used characters, only a very small portion (less than 5%, based on observation of the sample documents used) contain vertical strokes that resemble those found in Korean scripts. The Chinese characters are more complex and dense, hence more strokes within each character cell tend to touch or intersect each other, resulting in fewer isolated vertical strokes.

It is worth noting that there are some structural differences between the vertical strokes in Chinese documents and those in Korean. For example, most vertical strokes in Chinese have a “hook” towards the left at the bottom part of the stroke, while the vertical strokes in Korean script are very “smooth” at the same position of the strokes.

However, our experimental result shows that the significant “proportional difference” of such vertical strokes in documents of these two languages is sufficient to differentiate them in most cases.

Japanese Script

There are several groups of characters in Japanese script: Kanji, Katakana and Hiragana. Katakana and Hiragana have small sets of characters (around 160) and most of them do not contain similar vertical strokes to those in Korean script. Characters in the Kanji group were derived from ancient Chinese characters and the shape of vertical strokes is very similar to those in Chinese.

The following describes the techniques and procedure of extracting this feature, which is based on locating isolated vertical strokes in scanned images of documents

in these three oriental languages.

Intuitively, a few criteria need to be satisfied for a connected component to qualify as a “Korean-style” vertical stroke.

1. Overall Slimness: As the name implies, a vertical stroke’s height should be greater than its width.
2. Relative Height: The component needs to extend through most of the entire height of the text line.
3. Head-off Slimness: In many cases (depending on the font), Korean vertical strokes have left-pointing “heads” at their top-most position. This head significantly adds to the overall width of the vertical stroke. Thus, we cannot apply a very strict requirement to criterion 1. We need to remove the head, and then consider the head-off slimness for the remainder. The head-off height should be much more than its width.
4. Non-Symmetry: After the head-cutting process, a vertical stroke cannot have horizontal bar(s) extending to both left and right.
5. Horizontal-bar limitation: A genuine vertical stroke has at most two horizontal bars with a relatively short width, and such horizontal bars are always located around the central part of the vertical stroke.

The following is the procedure we used to locate Korean-style vertical strokes. Usually, Korean vertical strokes have a small slanted portion at the top and some vertical strokes also contain one or two horizontal bars. These horizontal bars can be located at the left or the right side of the vertical stroke. Some examples of Korean vertical strokes are shown in Figure 36.

그 방에서의 우리의 생활은 지지부진 하게 하루하루가 지나가고

Figure 36: Korean text line illustrating different shapes of vertical strokes

In searching for the Korean long vertical strokes, we apply the following steps:

1. For a text line, locate its 8-connected components.

2. Assume the connected component has height h and width w . We apply criteria 1 and 2 in comparing its height and width, and also determine its height relative to the text-line height. If:

$$\begin{aligned} h/w &\geq 2.2 \text{ and} \\ h &\geq 0.6 * \textit{textline_height} \end{aligned} \tag{17}$$

then go to step 3. otherwise this connected component is not a Korean vertical stroke.

3. Due to the existence of the slanted portion at the top of Korean vertical strokes. the ratio of the height to width of this vertical stroke is smaller than the one with a simple vertical.

We apply criterion 3. The top portion, i.e. the top 1/8 of this connected component is removed and the height to width ratio of the remainder is used to determine whether it qualifies as a long vertical stroke without any horizontal bars. If it does. then this ratio would be a large value (e.g. 5 – 7). Otherwise. this component is not a Korean vertical stroke, or it may be a vertical stroke containing one or two horizontal bars which needs further screening.

4. For those components with the top portion cut-off and are not qualified to be Korean vertical strokes in step 3, we further check to see if they belong to the vertical strokes containing one or two horizontal bars. The horizontal bars of Korean vertical strokes cannot extend to both sides, and they should be relatively short.

We apply criterion 4. As shown in Figure 37, two points are found when we use the cutting line to remove the top 1/8 of the vertical stroke candidate. They are the left and right most points of the vertical stroke candidate on the cutting edge. From the cutting line, we draw a left stem line that extends a small distance (1/2 of the distance between the two points) leftward beyond the left point. Similarly we draw the right stem line from the cutting line right point. These are tolerances to allow for small protrusions on the vertical stroke.

The rule is that a vertical stroke candidate (after removing the top portion) can extend beyond at most one of the stem lines, because Korean vertical strokes do not have horizontal bars on both sides.

In this step, the vertical stroke candidate extending beyond both left and right stem lines are eliminated. Otherwise, they will be further screened in step 5.

5. This step checks the number of horizontal bars belonging to the component. From the previous step, we know at which side the bar is located. Here we count the number of horizontal bars. If this number exceeds 2, then this component is not a Korean stroke; otherwise this component is a Korean stroke containing one or two horizontal bars.

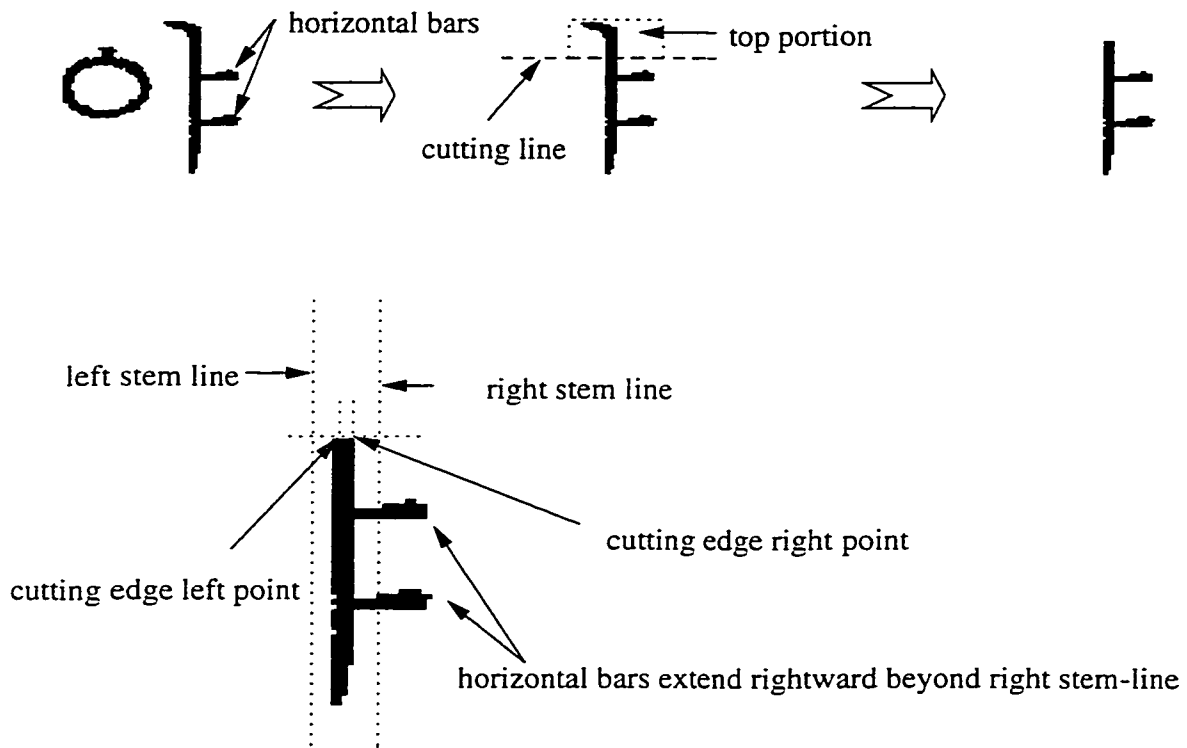


Figure 37: Non-symmetry of Korean vertical stroke

Through experimentation, we have established that isolated longer vertical strokes occur with greater frequency in Korean characters than in Katakana, Hiragana and Kanji characters.

4.4 Identification

4.4.1 Feature Vector Construction

We apply our feature extraction methods to each character cell. By counting the frequency of occurrences of the above proposed features, we accumulate the total frequencies of the complex structure, Korean “circle” and long vertical stroke in character cells, and obtain three values. The three values are then divided by the number of character cells in the document and normalized to fall within the range of 0 to 100. The three normalized values are then used as the document’s feature vector in the identification stage. These features are denoted by C(complex structure), K(Korean “circle”) and V(Korean vertical stroke), respectively, in the rest of this thesis.

4.4.2 Identification

We separate our images into training and testing sets. The training samples are randomly selected. In our database, there are *190* Chinese, *94* Japanese and *164* Korean document images. Of these, *76* Chinese, *45* Japanese and *58* Korean documents were chosen as the training samples. Figure 38 shows the feature distributions of these training samples, while Tables 6, 7 and 8 list the C, K, V values of Chinese, Japanese and Korean training samples.

Classification of the test samples are performed by two methods: by using the ranges of C, K and V values for the training sets, and by clustering. The results are presented in the next section.

The K-means clustering algorithm is used to generate cluster centers, or codebooks of the training data. Assume we want to generate M vector clusters:

1. Begin with one cluster with center at the centroid of the entire set of training vectors.
2. Double the size of the cluster by splitting each current cluster center \mathbf{y}_n according to the rules:

$$\begin{aligned} \mathbf{y}_n^+ &= \mathbf{y}_n(1 + \delta) \\ \mathbf{y}_n^- &= \mathbf{y}_n(1 - \delta) \end{aligned} \tag{18}$$

where n varies from 1 to the current number of clusters, and δ is a splitting parameter, usually $0.01 \leq \delta \leq 0.05$. In our case, δ is chosen to be 0.02.

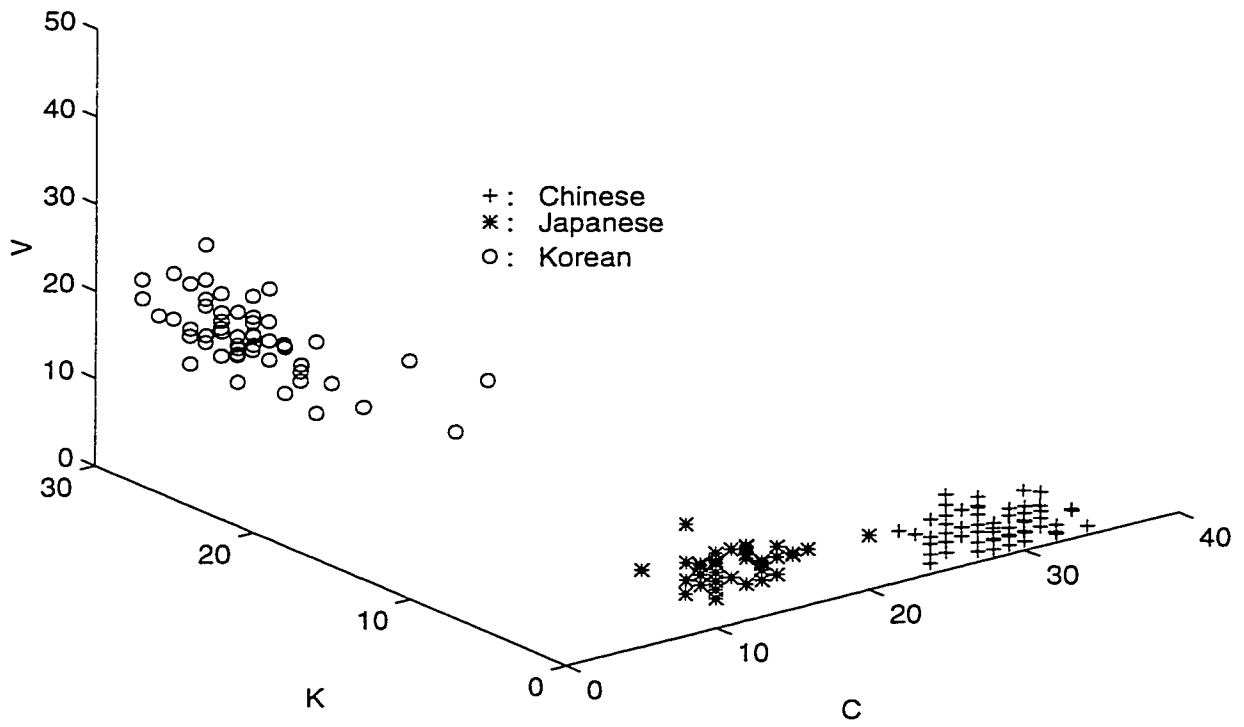


Figure 38: Feature vectors of the three training sets

Table 6: Feature vectors of Chinese training samples

No.	C	K	V	No.	C	K	V
1.	23	0	5	39.	29	1	1
2.	25	1	3	40.	29	0	3
3.	25	1	0	41.	29	1	1
4.	25	1	1	42.	29	0	1
5.	25	1	5	43.	29	2	1
6.	25	0	2	44.	29	0	3
7.	26	1	2	45.	29	2	3
8.	26	2	1	46.	29	0	2
9.	26	4	1	47.	29	2	5
10.	26	1	3	48.	30	0	2
11.	26	1	4	49.	30	2	1
12.	26	1	2	50.	30	0	1
13.	26	1	5	51.	30	0	2
14.	26	1	4	52.	30	0	1
15.	26	1	4	53.	30	0	2
16.	27	2	5	54.	30	0	5
17.	28	1	0	55.	30	1	1
18.	28	1	1	56.	30	1	4
19.	28	0	2	57.	30	0	2
20.	28	0	2	58.	31	1	2
21.	28	0	4	59.	31	1	1
22.	28	1	1	60.	31	2	2
23.	28	2	4	61.	31	1	2
24.	28	1	5	62.	31	1	3
25.	28	1	3	63.	31	0	4
26.	28	1	3	64.	32	0	1
27.	28	2	2	65.	32	1	2
28.	28	0	2	66.	32	2	1
29.	28	0	2	67.	32	1	5
30.	28	0	1	68.	32	1	2
31.	28	3	5	69.	32	0	2
32.	28	1	3	70.	32	1	1
33.	28	0	3	71.	33	1	0
34.	28	2	1	72.	34	1	2
35.	28	1	3	73.	34	3	1
36.	28	0	3	74.	34	0	1
37.	28	0	3	75.	34	4	2
38.	29	2	4	76.	35	2	1

Table 7: Feature vectors of Japanese training samples

No.	C	K	V	No.	C	K	V
1.	9	4	4	24.	13	4	3
2.	10	2	4	25.	14	3	5
3.	10	2	6	26.	14	2	6
4.	11	2	3	27.	14	1	5
5.	11	2	3	28.	14	1	3
6.	12	2	4	29.	15	2	4
7.	12	2	5	30.	15	2	3
8.	12	3	3	31.	15	3	5
9.	12	4	8	32.	15	3	4
10.	12	2	3	33.	15	2	3
11.	12	4	0	34.	15	2	3
12.	12	2	4	35.	16	2	2
13.	12	2	1	36.	16	1	5
14.	12	2	2	37.	16	2	4
15.	12	3	4	38.	16	2	2
16.	12	2	4	39.	17	2	4
17.	13	2	3	40.	17	2	4
18.	13	1	3	41.	17	3	4
19.	13	3	4	42.	17	2	4
20.	13	2	3	43.	18	2	4
21.	13	3	5	44.	18	2	4
22.	13	1	6	45.	21	1	5
23.	13	3	4				

Table 8: Feature vectors of Korean training samples

No.	C	K	V	No.	C	K	V
1.	1	22	19	30.	3	26	20
2.	1	24	19	31.	3	24	18
3.	1	27	19	32.	3	24	16
4.	1	25	15	33.	3	21	19
5.	1	25	19	34.	3	23	21
6.	2	22	19	35.	3	30	20
7.	2	19	19	36.	3	22	25
8.	2	23	17	37.	3	25	20
9.	2	22	20	38.	3	22	19
10.	2	23	22	39.	3	28	17
11.	2	29	19	40.	3	25	15
12.	2	25	22	41.	3	26	27
13.	2	26	23	42.	3	24	17
14.	2	26	17	43.	3	19	13
15.	2	20	15	44.	3	16	16
16.	2	24	19	45.	4	19	16
17.	2	23	14	46.	4	22	18
18.	2	21	18	47.	4	26	21
19.	2	25	17	48.	4	9	24
20.	3	20	17	49.	4	26	17
21.	3	26	23	50.	4	24	16
22.	3	22	19	51.	4	20	20
23.	3	23	19	52.	4	23	20
24.	3	22	19	53.	4	29	21
25.	3	25	19	54.	5	23	17
26.	3	25	19	55.	5	25	21
27.	3	24	16	56.	5	25	18
28.	3	20	16	57.	6	16	20
29.	3	25	19	58.	7	14	13

3. The best set of centroids for the split clusters is obtained by the following procedure:

- (a) For each training vector \mathbf{X} , assign it to the split cluster S_j using the Euclidean distance.

$$\mathbf{X} \in S_j, \text{ if } |\mathbf{X} - \mathbf{C}_j(p)| < |\mathbf{X} - \mathbf{C}_i(p)|$$

for all $i=1, 2, \dots, m$, where m is the number of split clusters,

S_j =set of samples whose cluster center is $\mathbf{C}_j(p)$.

Ties in the above expression are resolved arbitrarily.

- (b) Update centroids by using the training vectors assigned to each cluster.
(c) Compute the sum of the squared distances from all points in a cluster to its cluster center.
(d) Repeat steps (a), (b) and (c) until the difference between the distances of this iteration and the previous iteration falls below a preset threshold.

4. Iterate steps 2 and 3 until M vector clusters are generated.

In the splitting stage of the above clustering algorithm, one cluster is split into two, hence the number of generated clusters would be a power of 2. Based on the size of the training samples in our database and the results of some preliminary experiments, we have generated 4 clusters for each of the three languages. Hence altogether twelve centers are used to represent the training data, and they are listed in Table 9. Shown in Figures 39, 40 and 41 are the training data and cluster centers of Chinese, Japanese and Korean documents, respectively. The classification procedure of a given testing sample is a full search through all the cluster centers to find the "best" match. Assume there are M vector clusters with centers \mathbf{y}_m , $1 \leq m \leq M$, and the sample vector to be classified is \mathbf{v} . Then \mathbf{v} will be assigned to the class of cluster m^* if:

$$\begin{aligned} d(\mathbf{v}, \mathbf{y}_{m^*}) &< d(\mathbf{v}, \mathbf{y}_m) \\ 1 &\leq m \leq M \\ m &\neq m^* \end{aligned} \tag{19}$$

Table 9: Twelve cluster centers

Language	C	K	V
Chinese	26.57	0.96	2.00
Chinese	28.14	1.00	3.86
Chinese	30.24	0.62	1.67
Chinese	33.09	1.55	1.64
Japanese	11.73	2.27	2.93
Japanese	12.64	2.73	5.09
Japanese	15.17	1.92	3.58
Japanese	17.86	2.00	4.14
Korean	3.70	17.20	16.90
Korean	2.95	22.47	18.00
Korean	2.56	25.17	18.22
Korean	3.00	26.09	22.18

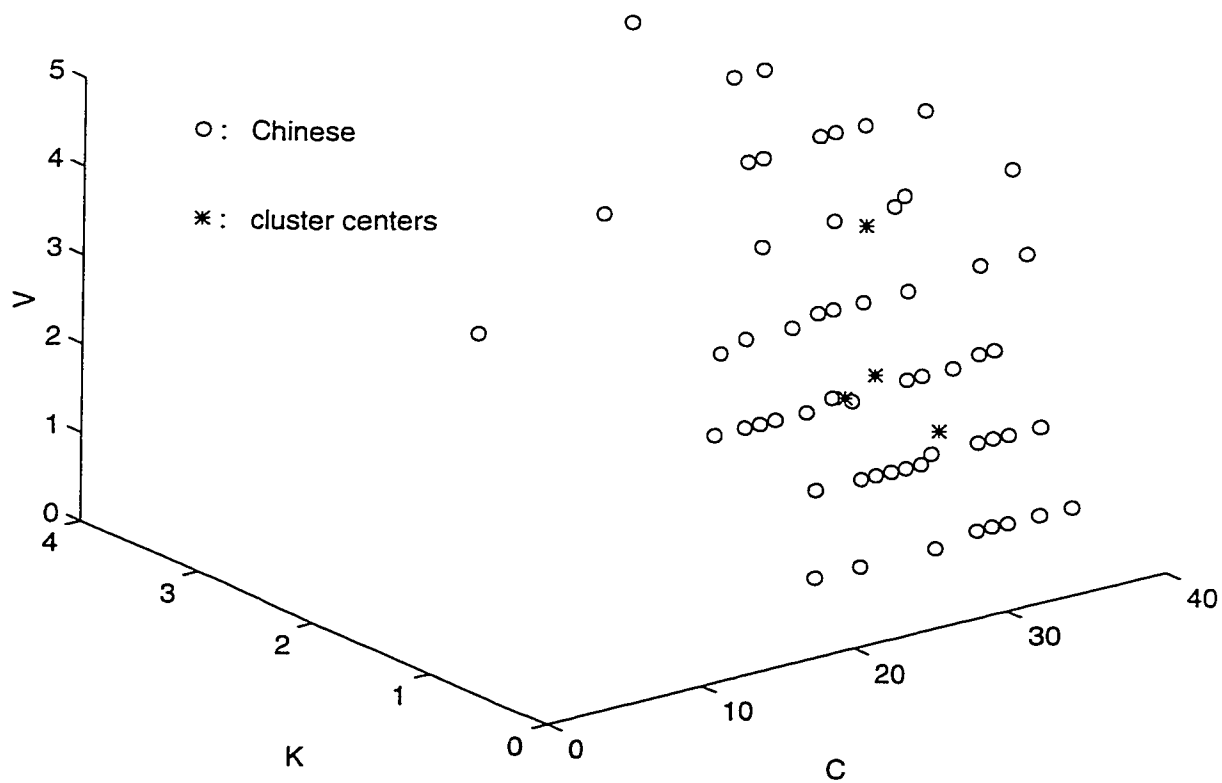


Figure 39: Chinese training data and their cluster centers

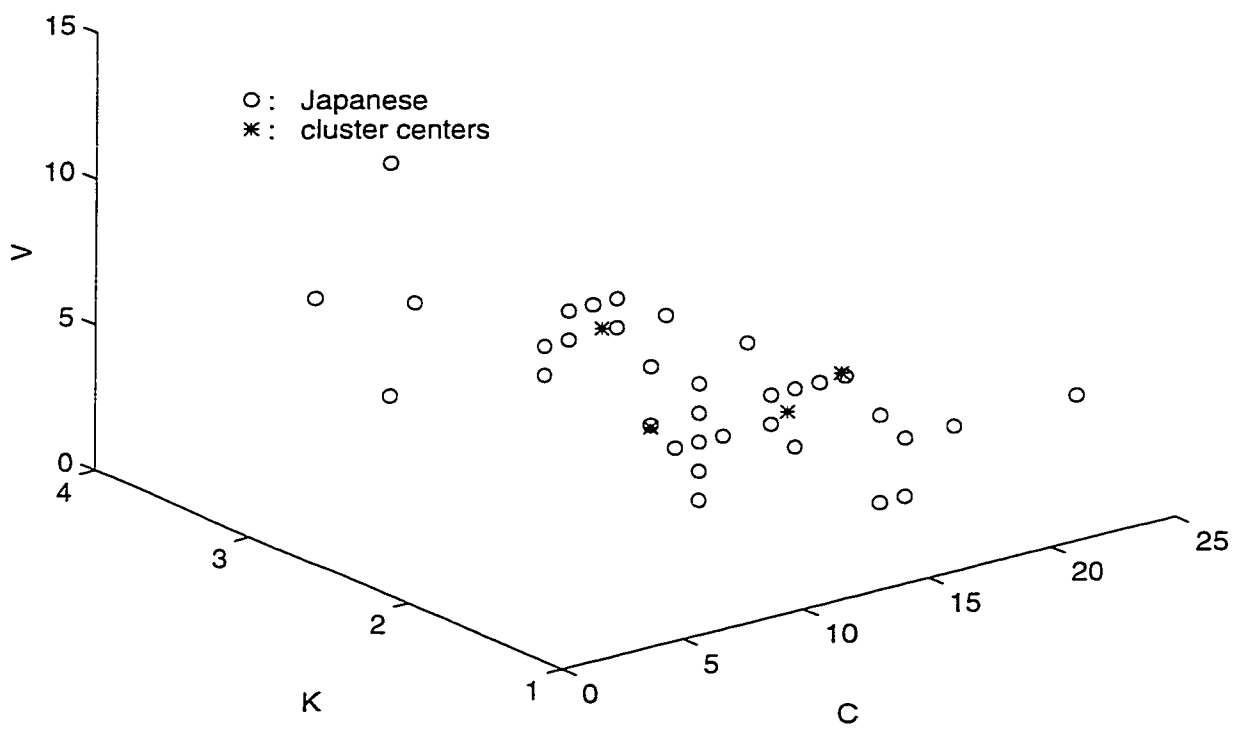


Figure 40: Japanese training data and their cluster centers

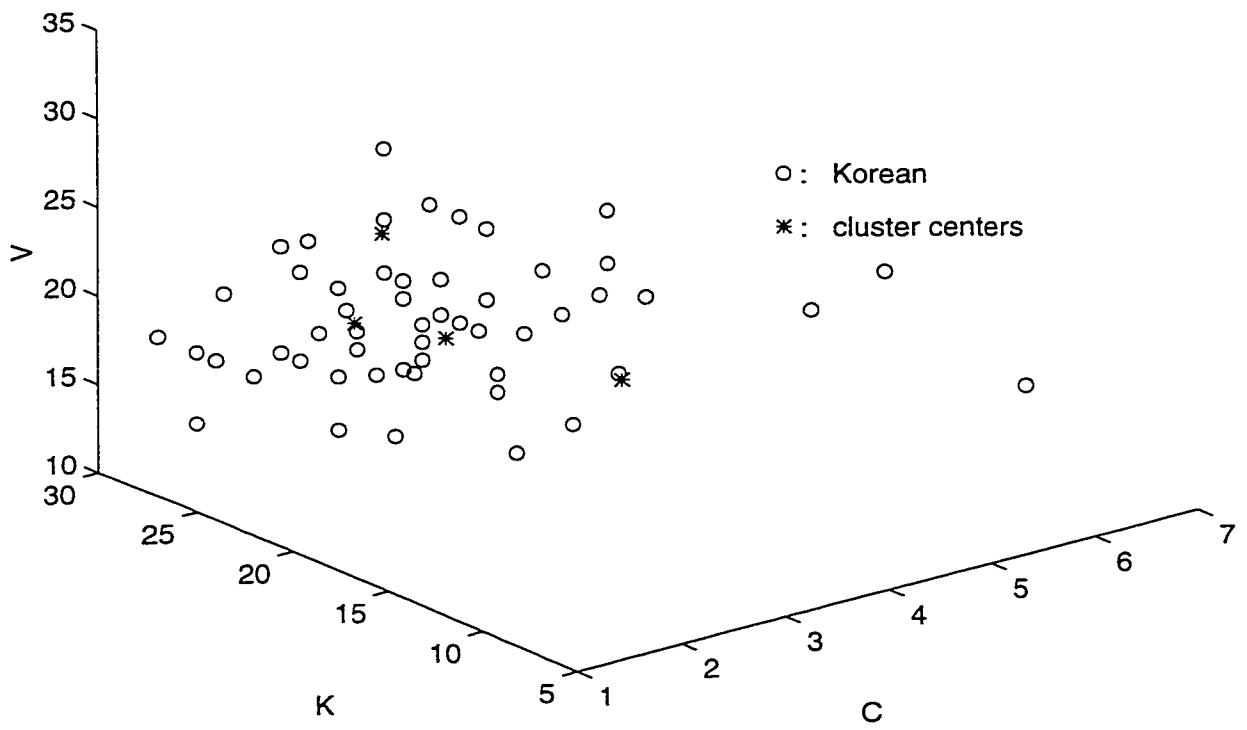


Figure 41: Korean training data and their cluster centers

4.4.3 Rejection Criteria

A test sample is considered closest to the class of cluster m^* based on formula (19). In addition, we compare the distances between the input and all other cluster centers with the minimum distance. If the difference between the minimum and any other distance is within a pre-defined value (in our case, we choose 0.2) and these two clusters represent different classes, this input is rejected because it is close to two different classes; otherwise the input is assigned to the class of cluster m^* .

4.5 Experimental Results

At present, our differentiation of the three oriental language scripts is based on the three proposed features extracted from one page of text. Each processed document image should contain at least 200 character cells, otherwise we consider it does not contain enough information to be identified.

4.5.1 Classification Results according to C, K, V values

Information from Tables 6, 7 and 8 indicates that Korean documents have “high” K and V values, while Chinese and Japanese have different range of C values. Based on the values of these three tables, we set the following rule to classify documents of the three languages:

1. If $K \geq 9$ and $V \geq 13$, then it is identified as Korean; otherwise
2. If $C \geq 23$, then it is classified as Chinese; if $C \leq 21$, it is Japanese; otherwise reject.

Applying this rule to the testing data, we obtained the classification results shown in Table 10, while the confusion matrix is given in Table 11.

4.5.2 Classification Results from Clustering

In an effort to improve classification results, we also used the clustering algorithm described in Section 4.4.2. The classification results and confusion matrix are shown in Tables 12 and 13 respectively.

Table 10: Results of oriental language classification by using C, K and V values

Language	# Samples	Not processed	Recognition (%)	Error (%)	Reject (%)
Chinese	114	1	94.69	4.43	0.88
Japanese	49	0	95.92	0.00	4.08
Korean	106	1	93.33	6.67	0.00

Table 11: Confusion matrix when using C, K and V values

	Chinese	Japanese	Korean	Reject
Chinese	107	5	0	1
Japanese	0	47	0	2
Korean	0	7	98	0

Table 12: Results from clustering using C, K and V features

Language	# Samples	Not processed	Recognition (%)	Error (%)	Reject (%)
Chinese	114	1	94.69	4.43	0.88
Japanese	49	0	97.96	0.00	2.04
Korean	106	1	97.14	1.91	0.95

Table 13: Confusion matrix from clustering using C, K and V features

	Chinese	Japanese	Korean	Reject
Chinese	107	5	0	1
Japanese	0	48	0	1
Korean	0	2	102	1

Chinese documents tend to be misclassified into Japanese when they are in Chinese Kai font because the strokes in this font are smooth and do not touch each other, which significantly reduces the number of complex structures. Korean documents are difficult to recognize when “ellipses” are used, because our Korean “circle” detection algorithm cannot handle this case when these “ellipses” resemble rectangles more than Korean “circles”.

4.5.3 Comparison of Clustering Results from Using Two Features

The above decision was made by using the combination of three features. Here we compare the results by using only two features. The three sets of results generated by different two-feature combinations are given in Tables 14, 16 and 18. The confusion matrices are also shown in Tables 15, 17 and 19. In Table 14, it is shown that one more Japanese document is misclassified into Chinese and several Korean documents are rejected when the vertical stroke is not considered. When the complex structure is not considered, the distinction between Chinese and Japanese becomes difficult and the recognition rates of these two languages decrease, as shown in Table 16. When the Korean “circle” feature is not considered, one more Japanese document is misclassified as Korean and one more Chinese document is rejected, as seen from Table 18. From the above results, we can draw the following conclusions:

- Complex structure is important for separating Japanese texts from Chinese.
- Long vertical stroke is important for differentiating Korean from Chinese and Japanese texts.
- Korean “circle/ellipse” is necessary for the separation of Korean from Chinese and Japanese texts.

4.5.4 Analysis of Results

When all three features are used in clustering, five Chinese testing samples are classified as Japanese documents. All but one of them are printed in Chinese Kai font. As the strokes in this font are smooth and do not touch each other, there are fewer

Table 14: Results of clustering from using C and K features

Language	# Samples	Not processed	Recognition (%)	Error (%)	Reject (%)
Chinese	114	1	95.58	4.42	0.00
Japanese	49	0	95.92	2.04	2.04
Korean	106	1	93.33	1.90	4.76

Table 15: Confusion matrix from using C and K features

	Chinese	Japanese	Korean	Reject
Chinese	108	5	0	0
Japanese	1	47	0	1
Korean	0	2	98	5

Table 16: Results of clustering from using K and V features

Language	# Samples	Not processed	Recognition (%)	Error (%)	Reject (%)
Chinese	114	1	82.30	11.50	6.19
Japanese	49	0	44.90	26.53	28.57
Korean	106	1	97.14	1.90	0.95

Table 17: Confusion matrix from using K and V features

	Chinese	Japanese	Korean	Reject
Chinese	93	13	0	7
Japanese	13	22	0	14
Korean	0	2	102	1

Table 18: Results of clustering from using C and V features

Language	# Samples	Not processed	Recognition(%)	Error (%)	Reject (%)
Chinese	114	1	93.81	4.42	1.77
Japanese	49	0	97.96	2.04	0.00
Korean	106	1	97.14	1.90	0.95

Table 19: Confusion matrix from using C and V features

	Chinese	Japanese	Korean	Reject
Chinese	106	5	0	2
Japanese	0	48	1	0
Korean	0	2	102	1

complex structures in this font, leading to the misclassification of these images into Japanese. Figure 42 shows one such sample. The misclassification of the other sample is due to poor quality and the existence of many broken strokes.

The Korean documents misclassified as Japanese are due to problems with the Korean “circle”. Most of the Korean “circles” in these documents cannot be detected by using our Korean “circle” extraction method. The ellipses in one document look more like rectangles than Korean “circles”, as shown in Figure 43. For the other document shown in Figure 44, if we carefully inspect the inner contours of the *circles*, we can see that sharp turns exist, which significantly reduced the number of Korean “circles”.

牌场终于结束了，痛快并未消退，接着的是吃。赢了的，反正是平白赢的，吃，输了的，能输起自己还吃不起？吃。数瓶的啤酒和一只烧鸡下肚了，饱嗝儿打过，吸一根烟吧，深深地吸下肚，长长地又吐出来，突然间感到了一切都是空的，都是无聊，这一夜就这么过去了，新的太阳即将出来，烦恼的明日还得烦恼，愁苦的明日还得愁苦，即使在这天欲明未明之际回家去，那老婆会给开门吗？

来时带上了愁苦烦恼和一揽子的百无聊赖要埋葬在牌场上，如今丢光了零钱又背上了愁苦烦恼和一揽子的百无聊赖该回走了。回走了，满地的是被嘴唇遗弃的烟头，心里想着这里人玩了牌还是牌玩了人，口里却说：喂，几时得空，再玩吧。

Figure 42: Some Chinese Characters in Kai Font

상기인을 귀 재단의 해외연수자로 파견함에 있어 다음사항을 확인합니다.

1. 파견자에 대한 인사상, 급여상, 신분상등의 불이익처분을 하지 않을 것임.
(단, 현재 정규직원으로 근무하지 않는 자는 귀국후 근무토록 조치할 것임)
2. 연수와 관련된 각종 행정사항이 발생했을시에는 타업무에 우선하여 지체없이 처리할 것임.
3. 연수자의 연수기간이 종료되는 즉시 귀국토록 할 것이며, 본인서약서 및 해외 Post-Doc. 연수 지원신청 요령에 위배되는 사항이 있을 경우에도 소환책임 및 그에 수반되는 모든 불이익을 연수자 본인이 직접 책임지도록 할 것임.

Figure 43: Korean document containing ellipses

한국 산문에 이런 기사가 난 것을 보았다. 부동산으로 ‘벼락부자’가 된 한국인들이 LA에 있는 W대학에서 강의 수료증을 준다고 해서 1인당 300만 원씩 내고 와서 수료증이라는 것을 받고 보니, “우리 대학을 방문해 주셔서 감사하다”는 내용의 감사장이었다고 한다.

결국 이들 서울의 부동산 졸부들은 미국 대학 수료증을 받아주겠다는 사기꾼에게 속은 것이다.

미국 대학 강의 수료증이라는 것이 무엇인지 모르지만, 하여튼 돈만 가지고는 만족하지 못하는 한국 졸부들의 허영심을 이용한 사기극의 하나에 불과하다. 부동산 투기로

Figure 44: Korean document containing sharp turns in circles

Chapter 5

Summary and Future Work

5.1 Summary

The automatic identification of the language used on documents is both useful and necessary, especially when more and more documents are processed electronically. However, language classification has never been as easy as might be imagined. Currently, no general solution exists for this problem, and much research is needed in this area.

In this thesis, the work of differentiation is mainly focused on two aspects. For a document printed in one of 24 languages, its language category (European or oriental) is first determined. If it is in the oriental category, then its language is identified.

After numerous experiments with many new distinction features, we have discovered good features that are well suited for the language-category classification and the identification of oriental documents. They seem to be very effective and the results are promising.

The following is a summary of this thesis:

In chapter 1, we discussed the general concepts of language identification and surveyed related research, particularly the methods used by other researchers in the areas of our work.

Chapter 2 briefly introduced the preprocessing techniques adopted to enhance the quality of document images. The major steps are noise removal, and line segmentation. Preprocessing is important for the quality of the image to be used for subsequent processing and classification.

Chapter 3 presented the differentiation between European and oriental languages. Given more than 20 different European languages and 3 oriental ones, we have proposed a set of new distinctive features including: horizontal projection profile, height distribution histogram and enclosing structure of connected components. Experimental results indicate that these features are very effective.

In chapter 3, we also applied an existing method [Spi94] and demonstrated that this method cannot satisfactorily distinguish the two language categories because our European source contains Roman as well as Cyrillic scripts which was not the case for [Spi94].

Chapter 4 discussed the identification of documents that belong to one of the three major languages in the oriental language category: Chinese, Japanese and Korean. Based on analyses as well as numerous experimentation on document images in these three languages, we have established an approach to solve this problem. Our method makes use of the complexity of structure, Korean “circle” and vertical stroke statistics. K-means clustering algorithm is used to do the actual classification.

Chapter 5 outlines the major contributions of this thesis and summarizes our effective work in European/oriental document differentiation and the identification of documents in three oriental languages. The directions of possible future work are also presented.

5.2 Main Contributions of the Thesis

The major contributions of this thesis are research and discovery of new distinctive features, and applying these features to automatically classify electronically scanned text documents.

Based on research results, several new features are proposed and proved to be effective in language classification. Three of these features are applied to separate European documents from Oriental ones, and another three are used to identify documents printed in Korean, Japanese and Chinese.

Experiments show our method achieved satisfactory results (over 90% correct rate) when applied to seven hundred documents printed in more than 20 different languages.

5.2.1 Contribution 1: New Features to Separate European and Oriental Documents

Based on the study of the generalities of European scripts, the common characteristics of the three oriental languages and the distinction between these two language categories, we have extracted a few new distinctive features to differentiate documents printed in European languages from those in oriental ones. They are:

- Horizontal projection profile,
- Height distribution of connected components, and
- Enclosing structure of connected components' bounding box.

The goal of our features is to maximize the commonality of documents printed in the same language category and maximize the difference among documents belonging to different categories.

Experiments indicate such features are well-suited for this purpose.

5.2.2 Contribution 2: New Features to Identify Chinese, Japanese and Korean Documents

The identification of the three oriental languages is quite difficult because they have very large character sets and also the size of the character sets is still increasing.

By analyzing the visual characteristics of their characters, we have proposed a set of distinctive features to identify such documents. These features try to capture the differences between the characters that constitute the three languages. However, the fact that Japanese documents contain Kanji characters greatly complicates the work.

Experimentation has been carried out with various features, with the conclusion that the following features have been most effective:

- Complexity of structure,
- Korean "circles", and
- Long vertical stroke.

Experiments also indicate the results are promising.

5.3 Conclusion and Future Work

5.3.1 Conclusion

Language differentiation is a difficult on-going research topic. In this thesis, we proposed our methods for the differentiation between European and oriental documents, also for the identification of documents in three major oriental languages.

The new features and method used are proven to be effective and appropriate based on the test results of over seven hundred text documents printed in 24 different languages.

Our method has the following advantages:

- It works quite well in the processing of “mixed” documents containing a mixture of both language groups (which are quite common in technical documents), provided that the non-host language(s) content does not exceed the limit of about 20% of the whole document. Other researchers have not reported their results on this aspect.
- Our method has been developed to handle documents that might be written in any of 24 different languages. For example, our method works well on Cyrillic documents that do not possess the same characteristics as documents in Roman languages. Most existing methods do not handle so many languages.
- Our classification results are based on statistical features, hence it is not very vulnerable to noise, broken characters or touching components. This makes it robust for practical use.

However, our Korean circle detection method cannot separate Korean circles from ellipses and hence the recognition rate will decrease when ellipses are used in certain Korean fonts. Also, for the Chinese Kai font, the complex structure is not easy to detect as the strokes of this font are smooth and non-touching.

5.3.2 Future Work

Based on the work we have carried out and described in this thesis, future study can proceed in the following directions:

- As pointed out previously, our method of separating European documents from oriental ones is fairly robust, but errors tend to occur in the “mixed” documents when more than 20% of the considered components are in another language. Using more than one unit of 50-components would reduce the probability of encountering unusual occurrences of highly mixed scripts in a document [LDS98]. But when such occurrences are common within the document, multilingual OCR would be more appropriate.
- The algorithm for detecting complexity of structure is affected by certain Chinese character font, e.g. Kai font. As the strokes in this font are smooth and do not touch each other, fewer complex structures can be detected using our current method on this font. Searching for a new way of detecting complex structures in such fonts would increase the recognition rate of Chinese documents.
- The high frequency of “circle” components in Korean documents is used in our method to distinguish them from the other two languages. However, accurately determining a “circle” is not trivial, because these “circles” often are not really circles but close to “ellipses”, “rectangles” or other shapes. In addition, the coarse micro-appearance (as opposed to the smooth macro-appearance to human eyes) makes the detection algorithm very complex and less effective. Discovering a better way to identify those “circles” would definitely improve the recognition performance of Korean documents.

Bibliography

- [Aie91] D. V. Aiegler. *The Automatic Identification of Language Using Linguistic Recognition Signals*. PhD thesis, State University of New York at Buffalo, 1991.
- [Bat92] E. O. Batchelder. A learning experience: Training an artificial neural network to discriminate languages. Unpublished Technical Report, 1992.
- [Bee88] K. R. Beesley. Language identifier: A computer program for automatic natural-language identification of on-line text. In *Language at Crossroads: Proceedings of the 29th Annual Conference of the American Translators Association*, pages 47–54, October 1988.
- [Chu91] W. L. Chung. Hangeul and computing. In V.H. Mair and Y. Liu, editors, *Characters and Computers*, pages 146–179. ISO Press, 1991.
- [Cry94] D. Crystal. *An Encyclopedic Dictionary of Language and Languages*. Penguin Books, London, England, 1994.
- [CT94] W. B. Cavner and J. M. Trenkle. N-gram based text categorization. In *Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–169, Las Vegas, April 1994.
- [Eji94] K. Ejiri. Word frequency distribution in Japanese text. *Quantitative Linguistics*, 1(3):212–223, 1994.
- [GN70] G. Gallus and P. W. Neurath. Improved computer chromosome analysis incorporating preprocessing and boundary analysis. *Phys. Med. Biol.*, 15(3):435–445, 1970.

- [GS95] D. Guillevic and C. Y. Suen. Cursive script recognition applied to the processing of bank cheques. In *Proceedings of the Third International Conference on Document Analysis & Recognition*, pages 11–14. Montreal, Canada, August 1995.
- [HH] J. K. T. Huang and T. D. Huang. *An Introduction to Chinese, Japanese and Korean Computing*. World Scientific.
- [HKTK97] J. Hochberg, P. Kelly, T. Thomas, and L. Kerns. Automatic script identification from document images using cluster-based templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):176–181. February 1997.
- [Jai89] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall Information and System Sciences Series, New Jersey, 1989.
- [KK96a] H. J. Kim and P. K. Kim. On-line recognition of cursive Korean characters using a set of extended primitive strokes and fuzzy functions. *Pattern Recognition Letters*, 17:19–28. 1996.
- [KK96b] H. J. Kim and P. K. Kim. Recognition of off-line handwritten Korean characters. *Pattern Recognition*, 29(2):245–254, 1996.
- [Kul91] S. Kulikowski. Using short words: A language identification algorithm. Unpublished technical report, 1991.
- [Lau93] K. R. Launde. The history of the Japanese character set and its encoding. *Computer Processing of Chinese and Oriental Languages*, 7(1):85–94. June 1993.
- [LDS98] L. Lam, J. Ding, and C. Y. Suen. Differentiating between oriental and European scripts by statistical features. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(1):63–79, 1998.
- [LNB96] D. S. Lee, C. R. Nohl, and H. S. Baird. Language identification in complex, unoriented, and degraded document images. In *Proc. Int. Association for Pattern Recognition Workshop on Document Analysis Systems*, pages 76–98, Malvern, Pennsylvania USA, October 1996.

- [LS97] L. Lam and C. Y. Suen. Application of majority voting to pattern recognition - an analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics*, 27(5):553–568, September 1997.
- [Mus65] S. Mustonen. Multiple discriminant analysis in linguistic problems. *Statistical Methods in Linguistics*, (4):37–44, 1965.
- [Nak80] A. Nakanishi. *Writing Systems of the World Alphabets, Syllabaries, Pictograms*. Charles E. Tuttle Company, Rutland, Vermont & Tokyo, 1980.
- [NBS97] N. Nobile, S. Bergler, and C. Y. Suen. Language identification of on-line documents using word shapes. In *Proc. 4th Int. Conf. on Document Analysis and Recognition*, pages 258–262, Ulm, Germany, August 1997.
- [New87] P. Newman. Foreign language identification: First step in the translation process. In *Proceedings of the 28th Annual Conference of the American Translators Association*, pages 509–516, Albuquerque, New Mexico, October 1987.
- [NS93] T. Nakayama and A. L. Spitz. European language determination from images. In *Proceedings of the Second International Conference on Document Analysis & Recognition*, pages 159–162, Tsukuba, Japan, October 1993.
- [RDS4] C. Ronse and P. A. Devijver. *Connected Components in Binary Images: The Detection Problem*. Research Studies Press, 1984.
- [SH98] C. Y. Suen and M. Hamanaka. Visual cues for automatic identification of languages. In *Proc. Vision Interface*, pages 365–372, Vancouver, Canada, June 1998.
- [SMRW98] C. Y. Suen, S. Mori, H. C. Rim, and P. S. P. Wang. Intriguing aspects of oriental languages. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(1):5–29, 1998.
- [Spi94] A. L. Spitz. Script and language determination from document images. In *Proceedings of the 3rd Annual Symposium on Document Analysis & Information Retrieval*, pages 11–13, Las Vegas, USA, April 1994.

- [SS94] P. Sibun and A. L. Spitz. Language determination: Natural language processing from scanned document images. In *Proceedings of the 4th ACL Conference on Applied Natural Language Processing*, pages 15–21. Stuttgart, Germany, October 1994.
- [Str93] N. W. Strathy. A method for segmentation of touching handwritten numerals. Master's thesis, Concordia University, 1993.
- [Sue92] C. Y. Suen. Processing of Chinese and oriental languages. In A. Kent and J. G. Williams, editors, *Encyclopedia of Computer Science and Technology*, pages 333–347, Marcel Dekker, Inc., New York, 1992.
- [Wan88] P. S. P. Wang(ed.). *Intelligent Chinese Language, Pattern, and Speech Processing*. World Scientific Publishing Co., Singapore, 1988.
- [WWC82] F. M. Wahl, K. Y. Wong, and R. G. Casey. Block segmentation and text extraction in mixed text/image documents. *Computer Graphics and Image Processing*. 20:375–390, 1982.