A Text Conversion System Between Simplified and Complex Chinese Characters Based on OCR Approaches

Chun-Jen Lee (李俊仁)† Keh-Hwa Shyu (徐克華)†‡
Eng-Fong Huang (黃英峰)† Bor-Shenn Jeng (鄭伯順)†

† Telecommunication Labs.
Ministry of Transportation and Communications
P.O. Box 71, Chung-Li, Taiwan, R. O. C.
Fax: 886-3-4904464 Email: cjlee@twnmoctl.bitnet

‡ Institute of Computer Science and Electronic Engineering
National Central University, Chung-Li, Taiwan, R. O. C.

Abstract

An automatic conversion system capable of translating text between simplified and complex Chinese characters is presented in this paper. This OCR-based system demonstrates an efficient feature extraction algorithm to recognize either complex or simplified printed Chinese characters. A new postprocessing model is developed to facilitate meaningful conversion of words, as well as correction of character recognition errors. Experimental results show that the average recognition rates are about 99.2% and 95.3% for single-font and multi-font character recognition respectively. When tested with real documents printed in simplified Chinese characters, the recognition rate is 96.2% without using contextual information. Upon employing the proposed language model for postprocessing, the text conversion rate can be improved to 97.8%.

1 Introduction

Optical Chinese character recognition is an extremely challenging task due to the complex shapes and the large vocabularies in the Chinese characters. Recently, the technique of optical character recognition (OCR) in printed Chinese characters has made great improvement [1]-[4]. And several commercial products have been emerged in the market. As the optical Chinese character recognition technique gradually matured, recent interests have been focused on practical applications of these algorithms.

Since 1956, mainland China regime has been advocating the use of simplified Chinese characters while Chinese in Taiwan and other parts of the world continue to use the traditional, and more complex characters. It is not easy for people used to traditional Chinese characters to read and understand the simplified characters. The same is true for those who learned only simplified Chinese characters. Recently, the open-door policy of mainland China has prompted peoples from both sides of Taiwan Strait to escalate business and cultural interactions between each other. To expedite timely communication and close cooperation, there is an ever increasing demand for speedy and reliable text conversion of printed documents between the simplified and complex Chinese character systems. An automatic system which can translate to and from simplified and complex printed Chinese characters is not yet available.

In this paper, we will present our effort in developing a first automatic text recognition and conversion system which achieves this objective. First we developed an OCR subsystem which can accurately recognize either simplified or complex printed characters. However, converting between simplified and complex forms is not an one-to-one mapping. Different complex Chinese characters may be represented by the same simplified Chinese characters. Some Chinese words used in China and in Taiwan are also different. To solve this problem, we developed a novel Chinese language postprocessing model. Utilizing the contextual relations, our system is able to overcome the ambiguities arisen during text conversion, and will also help correcting mistakes made during the OCR process.

2 System Description

The proposed system as depicted in Fig. 1 consists of four modules: segmentation, feature extraction, character recognition, and postprocessing. To facilitate recognition and postprocessing, several tables and dictionaries are also required and shown in Fig. 1. The functions of these tables and dictionaries will be described later.

An optical scanner will scan the document and convert it into a binary image. The segmentation module [5] will segment the entire image into blocks and then classify each block into a text, graphic, or picture block. The text blocks are further segmented into individual character blocks subsequently. Due to space limitation, details of the segmentation algorithm [5] are omitted in this paper.

Once the printed character blocks have been properly segmented, the feature

extraction module will extract features from each individual character. In the recognition stage, the extracted features of each character image are matched to a feature database to recognize the character. The top ten candidates for each character image are kept for subsequent processing. During the postprocessing stage, correction, conversion, and word replacement are carried out simultaneously to yield the final conversion result.

3 Feature Extraction

Two sets of features are used in our OCR subsystem. One is the crossing counts feature and the other is the accumulated stroke distribution feature [2]. The former is used as a filter to roughly sieve characters dissimilar to the target one, and the latter is used to recognize the character. They are described in the following:

Crossing counts: A crossing count is obtained by counting the points at which the pixel value turns from 0 (white) to 1 (black) along horizontal or vertical raster scan lines. Eight features based on crossing counts are used in our system, which are the average vertical crossing count, the average horizontal crossing count, the horizontal crossing count of top 1/3 character image, the horizontal crossing count of bottom 1/3 character image, the vertical crossing count of right 1/3 character image, the vertical crossing count of left 1/3 character image, the number of vertical separable components, and the number of horizontal separable components.

Accumulated stroke distribution feature: To describe this feature, a Chinese character pattern "本" is demonstrated as an example and shown in Fig. 2. For each pixel, we calculate four direction run-lengths, which pass the pixel. That is the distance from this pixel to the two opposite boundary pixel of the object. Thus, four distances will be obtained for each pixel as shown in Fig. 3. They represent the lengths of the stroke going through each pixel along four respective directions. Since the dimension of the matrix obtained above is usually too large for pattern matching, the matrix size can be reduced to 8×8 lattices in order to make pattern matching easier and faster. To represent the stroke intensity in each direction of the lattices, all of the matrix elements in each lattice are added together. After the dimensionality is reduced, the four reduced matrices are merged into one matrix. The direction of the strongest stroke intensity is selected as the feature of the lattice. If both the horizontal and vertical distances are larger than half of the width and height of the character image respectively, another feature is chosed to represent it. The feature code is illustrated in Table 1. Each character needs 72 feature bytes in the template feature data base, as shown in Fig. 4.

The use of a symbol or a code to represent a feature can help to construct the accumulated features of a template in the learning process. Logical *OR* operator is applied to merge the stored bytes of these two feature codes bit by bit so that a new

Bit assignment	Meaning
00000001	No stroke
00000010	Vertical stroke
00000100	Stroke along 135° direction
00001000	Horizontal stroke
00010000	Stroke along 45° direction
00100000	Both horizontal and vertical stroke

Table 1: Symbol definitions and bit assignments.

feature code can be generated for the construction of the template.

In the learning phase, these multi-font characters are learned separately, and the features of each font are also extracted separately. These features are then combined into a single feature template, stored in the feature database, by logical *OR* operation. This single feature template can encompass all features of various styles. Therefore, this system can recognize Chinese characters of different styles at the same time.

4 Character Recognition

After the feature codes of the input characters have been generated, they are matching with the ones of the stored templates according to a minimum distance criterion. The matching procedure is described as following:

Step 1: The crossing counts features are first examined. The distance of the input feature with the stored feature is calculated by the following formula:

$$d_{1i} = \sum_{k=1}^{8} NOT(P_k \ AND \ a_{ik}) \tag{1}$$

where P_k is the input feature vectors, and a_{ik} is the ith template feature vectors.

Step 2: If the $d_{1i} \leq 1$, goto *Step 3*. Otherwise, goto *Step 1* and match the next template.

Step 3: Match the accumulated stroke distribution features. The distance calculation of formula is almost the same as (1).

$$d_{2i} = \sum_{k=9}^{72} NOT(P_k \ AND \ a_{ik}) \tag{2}$$

where P_k is the input feature vectors, and a_{ik} is the ith template feature vectors.

Step 4: If all templates are matched, the character code with the minimum distance is the recognition result. Otherwise, goto *Step 1*, match next template.

After the processing of the matching procedure, there are 10 candidates for each character image to be kept for subsequent postprocessing. If more than one candidates have the same distance, the most frequently used character is selected first.

5 Postprocessing

There are cases in which different complex Chinese characters would be mapped to a single simplified Chinese character. For example, the complex Chinese characters {'復','複','覆'} are simplified to the same simplified Chinese character '复'. Currently, 110 sets of such character mappings in the internal code mapping table have been collected in the system. Besides, there are many Chinese word pairs in different terms, but each with the same meaning also exits. Especially, some foreign words will be represented as different words in China and Taiwan. For example, the English word 'laser' has been heterogeneously translated into '激光' in China and '雷射' in Taiwan. A total of 112 word pairs are collected in our word conversion dictionary.

A postprocessing model is included in our OCR-based conversion system to solve the problems of OCR recognition errors and the ambiguities in text conversions. To illustrate it, an example is given in Fig. 5. An input text (in simplified characters) '发展激光技术' (To develop the laser technology) is shown in Fig. 5 (a). Some possible complex-character candidates of Fig. 5 (a) are demonstrated in Fig. 5 (b), whereas the possible word hypotheses of Fig. 5 (b) are indicated in Fig. 5 (c). The final (correct) target text '發展雷射技術' after postprocessing is listed in Fig. 5 (d).

In this example, at character level, we simply assume that the first character '发', the second character '展', the fourth character '光', and the last character '术' are exactly and correctly recognized. Character sets {'發','髮'} and {'術','朮'} are the corresponding complex-character sets of the first and last characters, respectively. The third character '激' has two candidates '激' and '滿', while the fifth character '技' has three candidates '按','技', and '波'. At word level, { '發展' (develop), '激光' (laser), '光波' (light-wave), '技術' (technology) } constitutes a set of several possible word hypotheses. At word-conversion level, the word '激光' should be converted into the word '雷射'.

To illustrate the strategies of achieving the automatic text conversion based on OCR approaches, they are described as follows.

Simplified-to-Complex Chinese Characters

After a text sentence is recognized by OCR, as the example mentioned above, their candidates can form a character lattice as shown in Fig. 5 (b). Our strategy for converting simplified Chinese characters into complex ones consists of three steps :

1. Every simplified character extends its corresponding complex characters by searching the internal code mapping table and adds them to the word lattice as shown in Fig. 6.

2. Referring to the word conversion dictionary, a word identification process carries out word matching on partial string of the character lattice. For those partial strings form words, their corresponding complex-character words will be added to the word lattice.

3. A statistical Markov language model which will be described in the next section is employed to select the most probable sentence hypothesis from this given word lattice.

Complex-to-Simplified Chinese Characters

In the reverse process, the text conversion from complex Chinese Characters to simplified ones also consists of three steps :

1. A Markov language model is directly applied to a given character lattice (namely OCR candidate lists) to select the most probable sentence hypothesis.

2. Check whether those words, identified in the previous step, are appeared in the word conversion dictionary or not. If so, a word conversion process occurs and replaces all those partial strings of the most probable sentence hypothesis with their corresponding simplified-character words.

3. The remaining complex characters of the most probable sentence hypothesis are converted into simplified characters by looking up the internal code mapping table.

6 A Markov Language Model

According to the contextual information, a statistical bigram Markov language model [6] is embedded in our postprocessing module to improve the text recognition rate and the text conversion accuracy. The purpose of language model is to find the most possible candidate with maximum likelihood probability for each character in a given image sentence S. Let '$s_1 s_2 \ldots s_N$' be the character image sequence of sentence S, '$c_1 c_2 \ldots c_N$' be one of S's character candidate sequence C, and '$w_1 w_2 \ldots w_M$' be one of C's corresponding word sequence W, where N is the length of sentence S and M is the number of words in the sentence S. A word can be a single character or more than one character. This is a problem of finding a maximum likelihood sentence hypothesis $\hat{C}$ among all sentence hypotheses $\{C\}$. Thereby, our language model is defined as follows :

$$\hat{C} = \arg\max_{C} P(C|S) \tag{3}$$

$$P(\hat{C}|S) = \max_{C} P(C|S) \tag{4}$$

To compute the probability $P(C|S)$, we can formulate $P(C|S)$ as the following equations :

$$\begin{aligned} P(C|S) &= P(c_1 c_2 \ldots c_N | s_1 s_2 \ldots s_N) && (5) \\ &\approx \prod_{i=1}^{N} P(c_i|c_{i-1}) P(c_i|s_i) && (6) \\ &\approx \prod_{i=1}^{M} P(w_i|w_{i-1}) \prod_{j} P(c_j|s_j) && (7) \\ &\approx P(W|S) \end{aligned}$$

Based on the fundamental principle of bigram Markov model, the probability $P(C|S)$ can be directly approximated by Eq.(6). Since a Chinese word is the smallest meaningful unit in Mandarin Chinese, a word-level Chinese bigram model is adopted as our language model. Thus, we can estimate the probability $P(C|S)$ by Eq.(7). From [7], the factor $\prod_j P(c_j|s_j)$ in Eq.(7) can be regarded as the product of OCR similarity scores of c_j for word w_i and represented as $\prod_j OCRscore_j$, so we have :

$$P(C|S) \approx \prod_{i=1}^{M} P(w_i|w_{i-1}) \prod_{j} OCRscore_j \tag{8}$$

Although we focus on word-based bigram model, the probability values $P(w_i|w_{i-1})$ are difficult to estimate and retrieve due to the huge number of possible word bigrams from a large corpus. Generally, a common dictionary may comprise more than 100,000 Chinese words. One way to circumvent this problem is to estimate the probability values $P(w_i|w_{i-1})$ by $P(c_i^F|c_{i-1}^L)$ as suggested from [8]. As a result, the probability $P(C|S)$ can be further expressed by the following equation:

$$\breve{P}(C|S) = \prod_{i=1}^{M} P(c_i^F|c_{i-1}^L) \prod_{j} OCRscore_j \tag{9}$$

where c_i^F is the first character of word w_i, and c_{i-1}^L is the last character of word w_{i-1}.

7 Experimental Results

The testing and training database are built in two fonts, Kai font and Ming font, and three different styles for each font. For each font and style, there are four sets of character images, two sets by scanning the original documents and two sets by scanning copied documents. Hence, there are 24 sets of character image in our database, 12 sets for training and 12 sets for testing. Each set consisting of 5401 daily-used characters. Our system is built on PC 486-33. The character images are scanned by Microtek scanner with 400dpi resolution. A large Chinese text database of 23 million characters is collected from *United Daily News*, and used as our training corpus. In this experimental system, we have built up a common dictionary that contains about 30,000 Chinese words with high frequency of occurrence. Experimental results show that the recognition rates are 99.2% and 95.3% for single-font and multi-font characters, respectively, with the speed of 300 characters per minute. For field try, 8 random selected documents printed in simplified characters are used as the outside testing data. Our system achieves a recognition rate of 96.2% without contextual postprocessing. Using the proposed language model for postprocessing, we can improve an overall accuracy rate to 97.8% including the text conversion and the recognition error correction. The partial image from one of the selected articles is shown in Fig. 7 (a), whereas the recognition results of Fig. 7 (a) is depicted in Fig. 7 (b). The conversion rate of Fig. 7 (b) is 98.6%, and this results can highlight the accuracy and flexibility of the text conversion system.

8 Discussion

For the practical implementation, perhaps, the attention is paid on the relationship between the terms $\prod_{i=1}^{M} P(c_i^F|c_{i-1}^L)$ and $\prod_j OCRscore_j$ in Eq.(9). In our experiments, two scoring functions $SF1$ and $SF2$ as well as their corresponding weighting factors λ_1 and λ_2 were incorporated into Eq.(9) to tune the likelihood function of $P(C|S)$. This modified form can be expressed as:

$$\tilde{P}(C|S) = \lambda_1 SF1(\prod_{i=1}^{M} P(c_i^F|c_{i-1}^L)) + \lambda_2 SF2(\prod_j OCRscore_j) \quad (10)$$

where $SF1$ and $SF2$ are the scoring functions of $\prod_{i=1}^{M} P(c_i^F|c_{i-1}^L)$ and $\prod_j OCRscore_j$ respectively.

Besides, the conditional probability $P(c_j|c_{j-1})$ is defined by the following simple relative frequency

$$P(c_j|c_{j-1}) \triangleq \frac{F(c_j, c_{j-1})}{F(c_{j-1})} \tag{11}$$

where F is the number of occurrences of the character string in its argument appeared in the given training corpus. Due to the limited size of the training corpus, the amount of occurrence in $F(c_j, c_{j-1})$ may be unreliable. Therefore, the smoothing or interpolation methods [9], [10] should be adopted to estimate the conditional probabilities $P(c_j|c_{j-1})$.

Another consideration is the vocabulary size of lexicon. From our experience, the larger lexicon one can collect, the more coverage of contextual information one may get. However, this will cause more ambiguities of word boundary decision and need more computation. A compromise solution to solve the problem is to build up a moderate size of lexicon which is composed of words with high frequency of occurrence. With regards to the analysis of errors for text conversion, we found that only about 10% conversion errors came from the uncollected words and about 80% conversion errors from the fault of the OCR subsystem.

To reduce the computation cost, the bigram Markov language model can be easily and effectively evaluated by the dynamic programming algorithm on the network structure of the given word lattice to find the best path. Moreover, we only apply the Markov language model on each marked character for finding the best candidate. A character is marked if it requires to be converted or belongs to OCR confusion sets. Obviously, though the results presented in the previous section are encouraging, only top-1 path is hard to achieve higher accuracy rate under the current strategy using only pure word-level knowledge source. The efficient search algorithms for finding the N best pathes have been proposed [11], [12] and successfully applied to many applications. It is advantageous to combine one of the N-best algorithms [11], [12] with our current framework first to produce a scored list of all the likely pathes. This multiple pathes is then rescored and filtered using other linguistic information including syntactic, semantic and pragmatic knowledge sources to arrive at the best single path.

9 Conclusions

In this paper, we present a complete automatic Chinese character recognition and conversion system which is capable of reading either complex font or simplified font printed document and translating text between each other efficiently. In addition, a new statistical bigram Markov language model embedded in our postprocessing module can improve the character recognition rate and the text conversion accuracy. Experimental results show that this system is very practical and useful for automatic Chinese text processing.

Acknowledgement

The authors would like to thank Dr. J. T. Wang, Director of Telecommunication Laboratories, Dr. I. C. Jou and Dr. Y. H. Hu for their invaluable advice and timely encouragement. We also thank the colleagues of the TL OCR group for their carrying out a certain part of the experiments.

References

[1] S. Kahan, T. Pavlidis, and H. S. Baird, "On the recognition of printed characters of any font and size", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 9, no. 2, pp. 274-288, 1987.

[2] B. S. Jeng, "Optical Chinese character recognition using accumulated stroke features", Optical Engineering vol. 28, no. 7, pp. 793-799. 1989.

[3] V. K. Govindan, "Character recognition - A review", Pattern Recognition, vol. 23, no. 7, pp. 671-683, 1990.

[4] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical review of OCR research and development", Proc. IEEE, vol. 80, no. 7, pp. 1029-1058, 1992.

[5] B. S. Jeng et al.,"A novel block segmentation and processing for Chinese-English document", Proc. SPIE, Visual Communications and Image Processing II, pp. 588-598, 1991.

[6] E. J. Yannakoudakis, I. Tsomokos and P. J. Hutton, "n-Grams and their implication to natural language understanding", Pattern Recognition, vol. 23, no. 5, pp. 509-528, 1990.

[7] H. J. Lee and J. S. Chang, "Language analysis model for Chinese character recognition", Report no. TL-81-5205, 1992.

[8] L. S. Lee, "Language modeling in Mandarin speech recognition with very large vocabularies", Report no. TL-82-5201, 1993.

[9] F. Jelinek and R. L. Mercer, "Interpolated estimation of Markov source parameters from sparse data", Pattern Recognition in Practice, E. S. Gelsema and L. N. Kanal, Eds., NorthHolland Pub. Co., Amsterdam, pp. 381-397, 1980.

[10] S. M. Katz, "Estimation of probabilities form sparse data for the language model component of a speech recognizer", IEEE Trans. Acoustics, Speech, Signal Processing, vol. ASSP-35, no. 3, pp. 400-401, Mar. 1987.

[11] R. Schwartz and S. Austin, "A comparison of several approximate algorithms for finding multiple (N-Best) sentence hypotheses", IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 701-704, 1991.

[12] F. K. Song and E. F. Huang, "A tree-trellis based fast search for finding the N best sentence hypotheses in continuous speech recognition", IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 705-708, 1991.

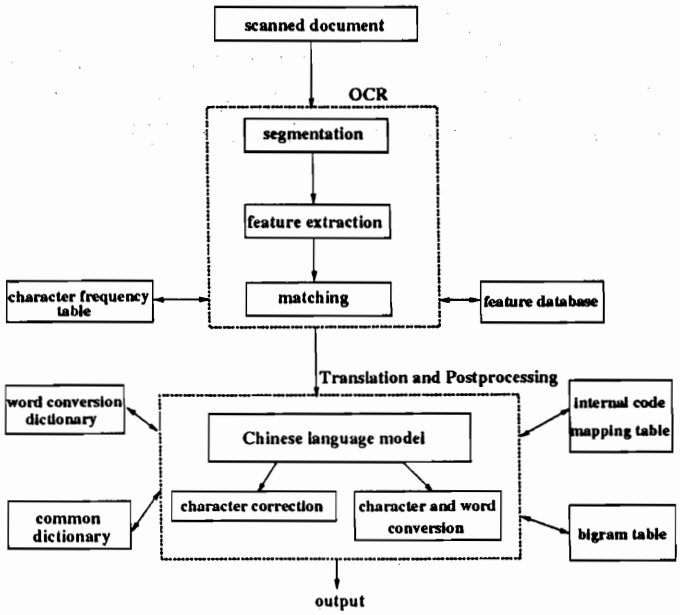

Figure 1: Block diagram of the system.

```
00000000000000000000000000
00000000000111000000000000
00000000000110000000000000
00000000000110000000000000
00000000000110000000000000
00000000000110000000001100
01111111111111111111111110
00000000001110000000000000
00000000011111000000000000
00000000011110100000000000
00000000110110110000000000
00000000110110010000000000
00000001100110001000000000
00000011000110001100000000
00000110000110000110000000
00001100000110000111000000
00011000000110000011111000
00100111111111111011100000
01100000000110000000100000
00000000000110000000000000
00000000000110000000000000
00000000000110000000000000
00000000000110000000000000
00000000000000000000000000
```

Figure 2: Dot-matrix of a Chinese character pattern '本'.

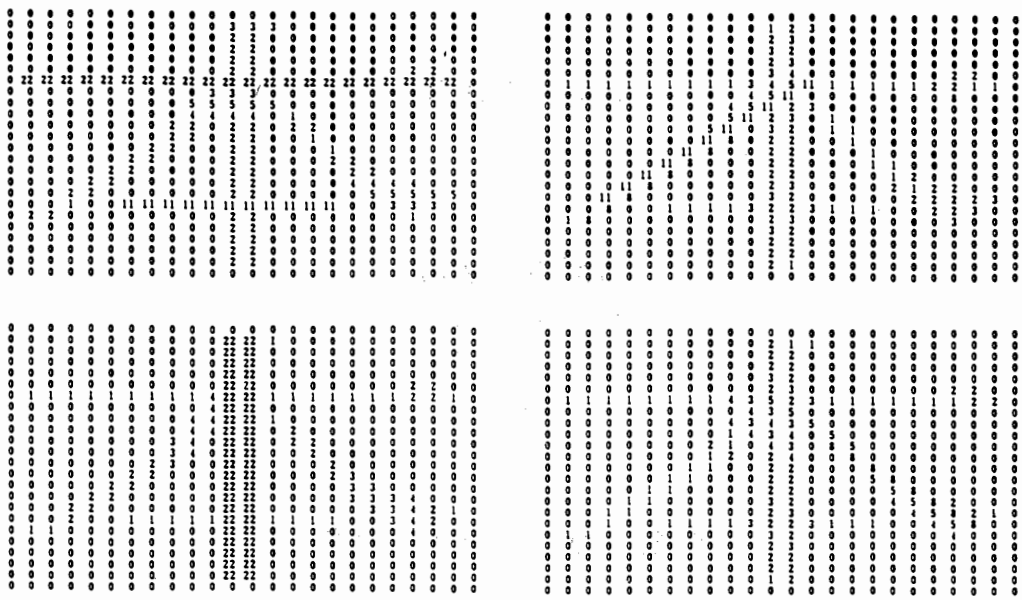

Figure 3: Distribution matrices for the strokes length of the character along four directions.

1	1	1	2	2	1	1	1
1	1	1	2	2	1	8	8
8	8	8	32	32	8	8	8
1	1	16	2	2	4	1	1
1	16	16	2	2	4	4	1
1	16	8	2	2	8	4	8
16	1	1	2	2	1	2	1
1	1	1	2	2	1	1	1

Figure 4: The accumulated stroke distribution feature matrix.

发展激光技术

(a)

發展激光按術
髮　滿　技朮
　　波

(b)

發展，激光，光波，技術

(c)

發展雷射技術

(d)

Figure 5: (a) An input text, (b) character candidates, (c) some word hypotheses, (d) the final result.

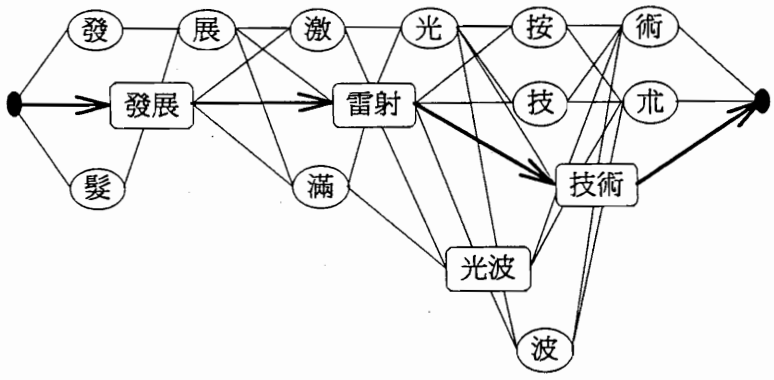

Figure 6: A word lattice.

蓬勃发展的高技术，对经济发展和人类社会的进步正在产生重大而又深刻的影响。美国、西欧各国、日本、苏联和东欧国家等都在竞相把开发高技术列为政治战略的主题。鉴于我国目前财力有限，技术水平不高，各种体制关系尚未理顺，还难以把高技术开发列为首要的任务。但也不能甘愿落后。应采取重点发展，有限目标的政策。我们将选择微电子和信息技术、生物技术、新材料技术等为主攻方向，并部署适当力量加强对航天技术、激光技术、海洋工程等高技术的研究与开发，逐步形成若干新兴产业。

(a)

蓬勃發展的高技術，對經濟發展和人類社會的進步正在產生重大而又深刻的影響。美國、西歐各國、日本、蘇聯和東歐國家等都在竞相把開發高技術列爲政治戰略的主題。鑒干我國目前財力有限，技術水準不高，各種體制關係尚未理順，還難以把高技術開發列爲首要的任務。但也不能甘願落後。應採取重點發展，有限目標的政策。我們將選擇微電子和資訊技術、生物技術、新材料技術等爲主攻方向，并部署適當力量加强對航太技術、雷射技術、海洋工程等高技術的研究與開發，逐步形成若干新興產業。

(b)

Figure 7: (a) A partial image, (b) the recognition results.