An Algorithm of Reducing the Number of Keystrokes for Chinese Phonetic Symbols

Cheng-Huang Tung and En-Yih Jean*

Department of Information Technology, National Pintung Institute of Commerce, Pingtung, Taiwan 900

e-mail: chdong@npic.edu.tw tel:886-8-7238700ext2700

*Department of Information Science, Aletheia University, Tamsui, Taipei, Taiwan 251

e-mail: ey.chien@msa.hinet.net tel:886-2-26212121

Abstract－ The input elements of a Chinese phonetic input method are 41 phonetic symbols, which are mapped to four rows of keystrokes. In this paper, we propose an algorithm for reducing the number of keystrokes for these phonetic symbols. Initially, a mapping function between 41 phonetic symbols and 4 rows of keystrokes is defined. By analyzing the phonetic strings of Chinese characters, we then define the criteria for prefix exclusion and context exclusion of two phonetic symbols. If two phonetic symbols have the property of prefix exclusion or context exclusion, we can map these two symbols to a same keystroke without ambigurity. In our algorithm, the user requires which keystrokes should be useless; that is, the requirement is that these phonetic symbols originally mapped to these keystrokes will be remapped to other keystrokes. Accordingly, we locate the phonetic symbols having the exclusion property with these phonetic symbols that should be remapped. Then, according to the measured exclusion property, the user manually determines to which keystrokes these phonetic symbols should be remapped. The experimental results show that the number of keystrokes can be reduced successfully after our algorithm has been performed.

keywords: phonetic symbols, keystroke, prefix exclusion, ontext exclusion

1. Introduction

The research about Chinese characters is very important and practical. There have been a great deal of recent efforts in solving the Chinese input methods [1-5]. Chinese characters are non-alphabetic and two-dimensional in structure[1]. There are thousands of commonly used Chinese characters, in which each character is ideographically different. Like English pronunciation, Chinese has the standard phonetic spelling system. Accordingly, the phonetic coding strategy[2-5] utilizes a phonetic spelling system to encode Chinese characters. Because the phonetic input method is very natural, it is it is undoubtedly very popular. There are totally 41 phonetic symbols, which are generally mapped to four rows of keystrokes. Howeveer, four rows of keystokes will encounter disadvantages for the phonetic input method. For example, if the input text mixes Chinese character and digits, such as “12 人共有 4352 元”， the input mode must be interchanged frequently between phoneme mode and alphabet mode.

In this paper, we aim to propose an algorithm to reduce the numbers of keystrokes for phonetic input. Initially, the user defines the keyboard mapping for 41 phonetic symbols, and the encountered keystrokes locate on four rows of keystrokes. Then, by analyzing the phonetic strings of Chinese characters, we can define the criteria for prefix exclusion and context exclusion of two phonetic symbols. If a phonetic pair has the property of prefix exclusion or context exclusion, the two phonetic symbols can be mapped to a same keystroke without ambigurity.

To reducing the number of keystrokes, the user can define which keystrokes must be useless. It means that the phonetic symbols mapped to these useless keystrokes will be remapped to other keystrokes. For each phonetic symbol mapped to a useless keystroke, we find out the phonetic symbols having the exclusion property with the specific phonetic symbol. Then, the user manually determines where the specific phonetic symbol can be remapped according to the exclusion property. By repeating the procedure, we can remap all the phonetic symbols originally mapped to the useless keystrokes. The experimental results show that the number of keystrokes can be reduced successfully from 41 to 30.

2. Prefix Exclusion of Phonetic Symbols

In this section, we define the criteria of prefix exclusion for two phonetic symbols. There are totally 5401 frequently-used Chinese characters. Let the 41 Chinese phonetic symbols be $P_1, P_2, \ldots, P_{41}$ shown in Table 1 where "ˊ", "ˇ", "ˋ" and "˙" are tone symbols. An effective phonetic sequence(EPS) is the phonetic sequence which some Chinese character is pronounced as. For example, the phonetic sequence "ㄐㄧㄣˋ" is an EPS, because it can be the pronunciation of character "進."

However, "ㄐㄣˋ" is not an EPS, because no characters can be pronounced as "ㄐㄣˋ."

Table 1. The Chinese phonetic symbols and pronunciation.

phonetic symbol	ㄅ	ㄆ	ㄇ	ㄈ	ㄉ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ
pronunciation	buh	puh	muh	fuh	duh	tuh	nuh	luh	guh	kuh
phonetic symbol	ㄏ	ㄐ	ㄑ	ㄒ	ㄓ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ
pronunciation	hu	gee	chi	shi	jr	chr	shr	re	dz	tz(e)
phonetic symbol	ㄙ	ㄚ	ㄛ	ㄜ	ㄞ	ㄟ	ㄠ	ㄡ	ㄢ	ㄣ
pronunciation	sz	ah	oh	uh	ai	a	au	oh(o)	an	un
phonetic symbol	ㄤ	ㄦ	ㄧ	ㄨ	ㄩ	ㄝ	ㄥ	ˊ	ˇ	ˋ
pronunciation	ang	uea	ea	wu	re	a	ong			
phonetic symbol	•									
pronunciation										

Next, we define that a keyboard mapping is a mapping function f between 41 phonetic symbols and keystokes; that is, we have $f(P_i)=(j,k)$ where (j,k) means that phonetic symbol P_i is mapped to the keystroke located at the jth row, kth column of the keyboard. In Fig. 1(a), we show the most popular keyboard mapping in Taiwan. Figure 1(b) shows the mapping function f for this keyboard mapping.

row \ column	1	2	3	4	5	6	7	8	9	10	11
1	ㄅ	ㄉ	ˇ	ˋ	ㄓ	ˊ	˙	ㄚ	ㄞ	ㄢ	ㄦ
2	ㄆ	ㄊ	ㄍ	ㄐ	ㄔ	ㄗ	ㄧ	ㄛ	ㄟ	ㄣ	
3	ㄇ	ㄋ	ㄎ	ㄑ	ㄕ	ㄘ	ㄨ	ㄜ	ㄠ	ㄤ	
4	ㄈ	ㄌ	ㄏ	ㄒ	ㄖ	ㄙ	ㄩ	ㄝ	ㄡ	ㄥ	

(a)

f(ㄅ)=(1,1), f(ㄉ)=(1,2), f(ˇ)=(1,3), f(ˋ)=(1,4), f(ㄓ)=(1,5),

f(ˊ)=(1,6), f(˙)=(1,7), f(ㄚ)=(1,8), f(ㄞ)=(1,9), f(ㄢ)=(1,10), f(ㄦ)=(1,11),

f(ㄆ)=(2,1), f(ㄊ)=(2,2), f(ㄍ)=(2,3), f(ㄐ)=(2,4), f(ㄔ)=(2,5),

f(ㄗ)=(2,6), f(ㄧ)=(2,7), f(ㄛ)=(2,8), f(ㄟ)=(2,9), f(ㄣ)=(2,10),

f(ㄇ)=(3,1), f(ㄋ)=(3,2), f(ㄎ)=(3,3), f(ㄑ)=(3,4), f(ㄕ)=(3,5),

f(ㄘ)=(3,6), f(ㄨ)=(3,7), f(ㄜ)=(3,8), f(ㄠ)=(3,9), f(ㄤ)=(3,10),

f(ㄈ)=(4,1), f(ㄌ)=(4,2), f(ㄏ)=(4,3), f(ㄒ)=(4,4), f(ㄖ)=(4,5),

f(ㄙ)=(4,6), f(ㄩ)=(4,7), f(ㄝ)=(4,8), f(ㄡ)=(4,9), f(ㄥ)=(4,10),

(b)

Fig. 1. (a) The most popular keyboard mapping for phonetic symbols. (b) The mapping function *f* for the keyboard mapping.

Then, we will define the criteria of preofix exclusion for honetic symbols P_i and P_j. We assume that EPS_m(=$P_{m_1}P_{m_2}\ldots$) and EPS_n(=$P_{n_1}P_{n_2}\ldots$) are two EPSs where $P_{m_k} = P_i$ and $P_{n_k} = P_j$. If we have $f(P_{m_p}) = f(P_{n_p})$ for all $1 \le p \le k-1$, then phonetic symbols P_i and P_j are not prefix-exclusive. For example, to check the prefix exclusion of phonetic symbols “ㄣ” and “ㄥ”, we analyze the two EPSs $P_{m_1}P_{m_2}P_{m_3}P_{m_4}$ (=“ㄐㄧㄣˋ”) and $P_{n_1}P_{n_2}P_{n_3}$ (=“ㄐㄧㄥ”). Because we have $f(P_{m_1}) = f(P_{n_1})$ (f(ㄧ) $=$ f(ㄧ)) and $f(P_{m_2}) = f(P_{n_2})$ (f(ㄐ) $=$ f(ㄐ)), phonetic symbols “ㄣ” and “ㄥ” are not prefix-exclusive. In the following, we propose the algorithm of determining whether two phonetic symbols are prefix-exclusive.

Algorithm 1: determine whether two phonetic symbols are prefix-exclusive

Input: two phonetic symbols P_i and P_j

Output: y/n

begin

  for each EPS $EPS_m(=P_{m_1}P_{m_2}\ldots)$ where $P_{m_k} = P_i$

  for each EPS $EPS_n(=P_{n_1}P_{n_2}\ldots)$ where $P_{n_k} = P_j$

    if $f(P_{m_p}) = f(P_{n_p})$ for all $1 \le p \le k-1$, return(n)

  return(y)

end

If two phonetic symbols are prefix-clusive, we can map these two phonetic symbols to a same keystroke. That is, after these two symbols are mapped to a same keystoke, all EPSs can be typed without ambiguity.

3. Context Exclusion of Phonetic Symbols

In this section, we will define whether phonetic symbols P_i and P_j are context-exclusion. The context exclusion is a weaker criteria comparied with prefix exclusion. For example, if we consider two EPSs "ㄓㄤ" and "ㄒㄧ" for phonetic symbols "ㄓ" and "ㄒ", the two EPSs make "ㄓ" and "ㄒ" not prefix-exclusive. However, if we show phonetic string "ㄓ(ㄒ)" on the screen for inputting the first symbol, we leave the first phonetic symbol undetermined; that is, the first symbol is possible to be "ㄓ" or "ㄒ." When the second symbol are typed as "ㄤ", we can determine that the first symbol is "ㄓ." Similarly, if the second symbol are typed as "ㄧ", we know that the first symbol is "ㄒ." After measuring all EPSs, we find that the two symbols "ㄓ" and "ㄒ" are not prefix-exclusive, but they are context-exclusive.

We assume that $EPS_m(=P_{m_1}P_{m_2}...)$ and $EPS_n(=P_{n_1}P_{n_2}...)$ are two EPSs where $P_{m_k}=P_i$ and $P_{n_k}=P_j$. If we have $f(P_{m_p})=f(P_{n_p})$ for all $1\le p\le k-1$ and $f(P_{m_{k+1}})=f(P_{n_{k+1}})$, then phonetic symbols P_i and P_j are not context-exclusive. For example, we have two EPSs "ㄐㄧㄣˋ" and "ㄐㄧㄥˋ". Since we have $f(P_{m_1})=f(P_{n_1})$ ($f(ㄧ)=f(ㄧ)$), $f(P_{m_2})=f(P_{n_2})$ ($f(ㄐ)=f(ㄐ)$) and $f(P_{m_4})=f(P_{n_4})$ ($f(ˋ)=f(ˋ)$), phonetic symbols "ㄣ" and "ㄥ"are not context-exclusive. In the following, we propose the algorithm to determine whether two phonetic symbols are context-exclusive.

Algorithm 2: determine whether two phonetic symbols are context-exclusive

Input: two phonetic symbols P_i and P_j

Output: y/n

begin

for each EPS $EPS_m(=P_{m_1}P_{m_2}...)$ where $P_{m_k}=P_i$

for each EPS $EPS_n(=P_{n_1}P_{n_2}\ldots)$ where $P_{n_k} = P_j$

if $f(P_{m_p}) = f(P_{n_p})$ for all $1 \le p \le k-1$

if $f(P_{m_{k+1}}) = f(P_{n_{k+1}})$ return(n)

return(y)

end

4. The Algorithm for reducing the number of keystokes

In this section, we will propose the algorithm of reducing the number of keystrokes for the phonetic symbols. According to algorithms 1 and 2, we can determine whether two phonetic symbols are prefix-exclusive or context-exclusive. If two phonetic symbols are prefix-exclusive or context-exclusive, we can map the two symbols to a same keystroke.

We define that the useless keystroke set, *UKS*, contains the keystrokes that should not be mapped by phonetic symbols. For example, if the contents of *UKS* is {(1,1),(1,2),…, (1,11)}, it means that the keystrokes at the first row will not be mapped by phonetic symbols. Initially, the 41 phonetic symbols will be mapped to the four rows of keystrokes, and then the user defines the contents of *UKS*. In the following, we propose the algorithm to leave these keystrokes in *UKS* useless.

Algorithm 3: leave the keystrokes in *UKS* useless

Input: *UKS* and a mapping function *f*

Output: an updated mapping function f

Begin

Repeat{

$S_{EX} = \{\}$

for each P_i where $f(P_i) \in UKS$

for each P_j where $f(P_j) \notin UKS$

call Algorithm 1 to check prefix exclusion

if P_i and P_j are prefix-exclusive, $S_{PE} = S_{PE} \cup \{(P_i, P_j, \mathrm{P})\}$

else

call Algorithm 2 to check context exclusion

if P_i and P_j are context-exclusive, $S_{EX} = S_{EX} \cup \{(P_i, P_j, \mathrm{C})\}$

if $S_{EX} \neq \{\}$

manually select one element $(P_i, P_j, \mathrm{P/C})$ from S_{EX}

remove $f(P_i)$ from *UKS*

update f by reassigning $f(P_i) = f(P_j)$

}until *UKS*=={} or no $(P_i, P_j, \mathrm{P/C})$ can be selected

end

5. Experimental results

In the experiments, we first define the mapping function and the contents of useless keytroke set *UKS*. As shown in Fig. 1, we have the most popular mapping function f for the phonetic keyboard mapping. Then, we define *UKS*={(1,1),(1,2),…, (1,11)}; that is, the keystrokes at the first row will be useless for the phonetic input method. Therefore, there are 11 phonetic symbols, "ㄅ", "ㄉ", "ˇ", "ˋ", "ㄓ", "ˊ", "˙",

“ㄚ”, “ㄞ”, “ㄢ” and “ㄦ”, that should be remapped.

In Fig. 2, we show the process of remapping 11 phonetic symbols. In the first three rows of Fig. 2, we show the exclusion result of measuring phonetic symbol “ㄅ” with other phonetic symbols. Property “P” and property “C” mean that symbol “ㄅ” and these symbols are prefix-exclusive or context-exclusive. Symbol “ㄅ” is prefix-exclusive with symbols “ㄝ”, “ㄟ” and “ㄥ” and are context-exclusive with symbols “ㄛ”, “ㄜ”, “ㄠ”, “ㄡ”, “ㄣ” and “ㄤ.” Because prefix exclusion is a stronger constraint than context exclusion, we will determine our selection from symbols with prefix exclusion property. The marked symbol, “ㄥ”, is selected manually by the user. Therefore, we have $f(ㄅ) = f(ㄥ)$. Similarly, we have $f(ㄉ) = f(ㄝ)$, $f(ˇ) = f(ㄍ)$, $f(ˋ) = f(ㄐ)$, $f(ㄓ) = f(ㄟ)$, $f(ˊ) = f(ㄗ)$, $f(˙) = f(ㄧ)$, $f(ㄚ) = f(ㄆ)$, $f(ㄞ) = f(ㄇ)$, $f(ㄢ) = f(ㄈ)$ and $f(ㄦ) = f(ㄌ)$. In Fig. 3(a), we show the keyboard mapping for the updated mapping function. In Fig. 3(b), we show the contents of the updated mapping function. Accordingly, the total number of keystrokes for 41 phonetic symbols is 30.

ㄅ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion																						C	C	P	P	C	C	C	C	P
ㄉ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion																						C	C	P	P	C	C	C	C	—
ˇ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P		P			—						—
ˋ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	P	P	P	P	P	P	—	P	P	P	P	P	P	P	P	P	P	P	P		P			—						—
ㄓ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion							—			—	C	C												—	P					—
ˊ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	P	P	P	P	P	P	—	P	P	—	P	P	P	P	P	P	P	P	P		P			—	—					—
˙																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	P	P	P	P	P	P	—	P	P	—	P	P	P	P	P	—	P	P	P		P	P		—	—					—
ㄚ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	C	C	C	C	C	C	—	C	C	—	C	C				—			—					—	—					—
ㄞ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	—	C	C	C	C	C	—	C	C	—	C	C				—			—					—	—					—
ㄢ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	—	—	C	C	C	C	—	C	C	—	C	C				—			—					—	—					—
ㄦ																														
symbol	ㄆ	ㄇ	ㄈ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ	ㄐ	ㄑ	ㄒ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄧ	ㄨ	ㄩ	ㄛ	ㄜ	ㄝ	ㄟ	ㄠ	ㄡ	ㄣ	ㄤ	ㄥ
exclusion	—	—	—	C	C	C	—	C	C	—	C	C				—			—					—	—					—

Fig. 2. The exclusion analysis of phonetic symbols “ㄅ”, “ㄉ”, “ˇ”, “ˋ”, “ㄓ”, “ˊ”, “˙”, “ㄚ”, “ㄞ”, “ㄢ” and “ㄦ.”

row \ column	1	2	3	4	5	6	7	8	9	10	11
1											
2	ㄆㄚ	ㄊ	ㄍˇ	ㄐˋ	ㄔ	ㄗˊ	ㄧ•	ㄛ	ㄓㄟ	ㄣ	
3	ㄇㄞ	ㄋ	ㄎ	ㄑ	ㄕ	ㄘ	ㄨ	ㄜ	ㄠ	ㄤ	
4	ㄈㄢ	ㄌㄦ	ㄏ	ㄒ	ㄖ	ㄙ	ㄩ	ㄉㄝ	ㄡ	ㄅㄥ	

(a)

f(ㄅ)=(4,10), f(ㄉ)=(4,8), f(ˇ)=(2,3), f(ˋ)=(2,4), f(ㄓ)=(2,9),

f(ˊ)=(2,6), f(•)=(2,7), f(ㄚ)=(2,1), f(ㄞ)=(3,1), f(ㄢ)=(4,1), f(ㄦ)=(4,2),

f(ㄆ)=(2,1), f(ㄊ)=(2,2), f(ㄍ)=(2,3), f(ㄐ)=(2,4), f(ㄔ)=(2,5),

f(ㄗ)=(2,6), f(ㄧ)=(2,7), f(ㄛ)=(2,8), f(ㄟ)=(2,9), f(ㄣ)=(2,10),

f(ㄇ)=(3,1), f(ㄋ)=(3,2), f(ㄎ)=(3,3), f(ㄑ)=(3,4), f(ㄕ)=(3,5),

f(ㄘ)=(3,6), f(ㄨ)=(3,7), f(ㄜ)=(3,8), f(ㄠ)=(3,9), f(ㄤ)=(3,10),

f(ㄈ)=(4,1), f(ㄌ)=(4,2), f(ㄏ)=(4,3), f(ㄒ)=(4,4), f(ㄖ)=(4,5),

f(ㄙ)=(4,6), f(ㄩ)=(4,7), f(ㄝ)=(4,8), f(ㄡ)=(4,9), f(ㄥ)=(4,10),

(b)

Fig. 3. (a) The keyboard mapping updated from themost popular keyboard mapping. (b) The updated mapping function f for the updated keyboard mapping.

6. Conclusions

In this paper, we have proposed an algorithm for reducing the number of keystrokes for phonetic symbols. Initially, the user defines a keyboard mapping between phonetic symbols and keystrokes. The algorithm can allow the user to define the set of useless keystrokes. According to the algorithm, we can remap the phonetics symbols mapped to these useless keystrokes to other keystrokes. In the experiments, we define the keyboard to the most popular phonetic keyboard mapping in Taiwan. The requirement is that the first row of keystrokes is useless. By applying the algorithm, we reduce the number of keystrokes from 41 to 30, and the number of rows of keystrokes from 4 to 3.

References

1. S. Kwong, H. Wong and Y.S. Yu(1991), An effective method for storing and retrieval of Chinese characters, Proc. of Int. Conf. on Information Engineering, Singapore, pp. 368-376.

2. S.K. Wan, H. Saitou and K.I. Mori(1982), Experiment on PINYI-HANZI conversion Chinese word processor, Computer Processing of Chinese and Oriental Languages, Vol. 1, No. 4, pp.213-224.

3. S.P. Zhou(1983), A proposal on Chinese PINYI encoding scheme, Proc. of Second National Conf. on Chinese Information Processing (in Chinese).

4. X. Cao and C.Y. Suen(1987), A new phonetic and ideographic coding Technique for Chinese information processing, Computer Processing of Chinese and Oriental Languages, vol. 3, No. 2.

5. H.Y. Gu(1994), A Chinese-character inputting system using a new type of phonetic keyboard and a compound Markov language model, Proc. of ROCLINGC, pp.253-262.