

흘림체 한글 필기의 온라인 원고 작성기 구현

권 오성, 권 영빈

중앙대학교 공과대학 컴퓨터공학과 컴퓨터비전 연구실

Implementation of A Continuous Cursive On-Line Hangul Handwriting Recognition System Based on the Boxed Style Pad

Oh-Sung Kwon, Young-Bin Kwon

Computer Vision Laboratory, Dept. of Computer Science & Engineering,
College of Engineering, Chung-Ang University

and

Center for Artificial Intelligence Research

요 약

본 논문에서는 한글의 자소간 흘림의 연속 필기를 허용하는 원고 작성기의 구현을 연구하였다. 이러한 온라인 한글 필기의 응용에서는 신속한 인식속도를 갖는 인식방법이 요구되며, 인식중에도 계속적인 필기가 가능하도록 하여 사용자에게 편의를 제공할 수 있어야 한다. 본 논문에서는 이와같은 요구사항을 만족시키기 위하여 스트링 정합방법에 기반한 신속한 인식 방법을 사용한다. 또한, 글자인식과 필기데이터 수집이 병행적으로 처리 되도록 구성됨으로써 원고작성시에 자유로운 필기동작이 가능하도록 하였다. 실험결과 50 명이 쓴 21,076자에 대하여 88.96%의 인식률을 제공하였으며, 제안하는 구현 방법이 원고 작성 응용에 적합하게 동작함을 알 수 있었다.

I. 서 론

키보드를 대신하여 보다 자연스런 형태의 컴퓨터 입력 수단에 관한 많은 연구와 노력이 있어 왔다. 그 중에서 전자평판(tablet)과 전자펜을 사용하는 방식은 현재까지 관심이 집중되어온 분야이다[6,14,17]. 본 논문에서는 원고지 형태의 화면위에 전자펜으로 한글을 필기하고, 필기된 한글을 인식하여 처리하는 온라인 원고작성기의 구현방법을 설명하고자 한다. 이러한 온라인 인식기를 이용한 응용에서 사용상의 불편함을 줄이기 위해서는 인식이 높은 인식률과 신속한 인식속도를 제공할 수 있어야 한다. 온라인 한글 인식의 경우, 자소내 흘림 필기 형태는 많은 연구가 있었고 몇 가지 제품들도 나오고 있으나, 자소간

홀림 필기의 경우는 심한 필기 변형과 자소분할의 어려움으로 인식과정이 복잡해져서 만족스런 인식률과 인식속도를 얻는 것이 쉽지 않다.

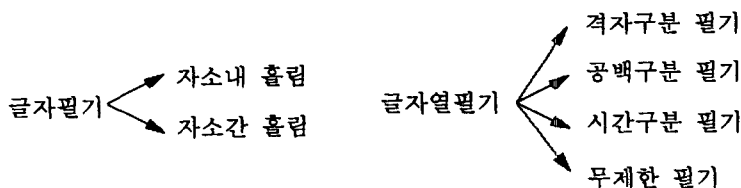
온라인 한글 인식에 관한 연구는 지속적으로 이루어져 왔으며, 대표적인 방법들을 살펴보면 다음과 같다. 우선, 획의 표준패턴을 정의하고 이를 조합하는 구조적인 방법을 사용하여 인식하고자 한 연구가 있었다[2]. 신경망을 이용한 경우의 연구로는 필기형태에 근거한 획(stroke)분류 테이블을 구성한 후 추출된 특징들을 획 인식 신경망의 입력으로 받아서 문자를 인식하도록 한 경우를 들 수 있다[4]. 이와 같은 방법들에서는 다양한 필기 방식때문에 필요한 획의 형태들을 미리 설정하여 놓기 어렵다는 문제를 지니고 있다. DP(dynamic programming)방법은 온라인과 오프라인의 한글을 포함한 문자인식에서 자주 사용되는 인식방법으로서 다양한 연구 결과들을 갖고 있다[3,6,13,14,16]. 일반적으로 DP방법은 신속한 인식속도를 갖기 어려우며, 탐색해야 할 공간을 줄이기 위하여 인식에 앞선 대분류 과정을 갖기도 한다. 통계적인 방법으로서 HMM(Hidden Markov Model)을 이용한 연구가 있으며, 이 방법에서는 한글의 제자원리에 따라 필기 네트워크를 구성하고, 확률적으로 최적의 경로를 찾기 위한 탐색을 하는 방식이다[5]. 이러한 통계적 방법은 홀림체 필기의 모델링에는 탁월한 효과를 갖지만, 계산상의 복잡도로 인한 인식시간의 문제점을 갖고 있다. 다른 방법의 연구로서는 인식과정에서 발생하는 획의 불명확성을 퍼지 멤버십으로 표현하여 해결하고자 한 경우의 연구를 들 수 있다[7].

한글의 자소간 홀림의 연속 필기를 허용하는 원고 작성기의 구현을 위해서는 신속한 인식속도를 갖는 인식방법이 요구되며, 인식중에도 계속적인 필기가 가능하도록 하여 사용자에게 편의를 제공할 수 있어야 한다. 본 논문에서는 첫번째 요구사항을 만족시키기 위하여 스트링 정합방법에 기반한 인식 방법을 사용한다. 이 정합방법은 스트링 편집과 비교순서함수에 의해 신속한 인식이 가능하다. 두번째 요구사항의 만족을 위해서는 원고 작성기의 글자인식과 필기 데이터 수집이 병행적으로 처리되도록 구성됨으로써 자유로운 필기동작이 가능하도록 하고 있다.

본 논문의 구성은 다음과 같다. II장에서는 원고작성기의 전체적인 구성을 설명하고, III장에서는 원고작성기에서 사용한 정합방법을 설명한다. IV장에서는 실험결과를 통하여 제안하는 인식방법의 효율성을 보이고, 끝으로 결론을 맺는다.

II. 원고작성기의 전체적인 구성

한글연속필기는 아래와 같이 구분될 수 있다. 이와같은 필기형태의 분류 중에서 자소간 홀림은 자소내 홀림을 허용하는 경우와는 달리 자소인식에 앞선 자소분할을 요구하며, 이러한 요구는 한글인식을 어렵게 하는 중요한 요인으로 작용하고 있다[1,3,5].



이러한 이유로 기존의 온라인 한글인식 연구는 주로 자소내 홀림 허용 필기 인식에서 이

루어져 왔으며, 최근들어 자소간 홀림의 연구도 활발히 수행되고 있다. 격자내 필기의 경우에 사용자는 원고를 작성하듯이 필기를 해야하는 제약이 따른다. 반면에, 이러한 유형은 글자열에서 글자분할을 위한 별도의 과정을 요하지 않는 용이함이 있다. 공백구분에 의한 필기는 글자와 글자 사이에 공백을 두면서 필기하는 형태로서 글자열로 부터 공백을 기준으로 글자를 분할하고 인식할 수 있으며, 이러한 필기형태 역시 필기시 주의를 요한다. 다른 필기 유형으로는 필기시 아무런 제약을 가하지 않는 무제한 필기를 들수 있고, 글자분할을 위해서는 보다 많은 연구가 이루어 져야 하는 유형이다.

인식기가 수행되는 시점에 따라 인식방법은 다음과 같이 3가지 유형으로 구분될 수 있다.

유형 1 : 한 문자 필기후 인식 수행

유형 2 : 문자열 필기후 인식 수행

유형 3 : 필기와 병행하여 인식 수행

이러한 구분중 유형 1의 방법은 한 문자의 필기를 마친 후, 인식기가 필기 종료 신호를 받아들여 인식을 수행하는 방식이다. 이 방식에서 사용되는 필기종료 신호는 주로 시간적인 정보 혹은 공간적인 정보로서 시간적으로 정의된 간격을 경과하거나 현재의 필기 공간을 벗어날 때 이러한 신호가 발생된다. 이러한 방법은 필기후 종료 신호가 발생되고 필기된 문자가 인식되기를 기다렸다가 다음 글자를 필기해야 하는 어려움이 따른다. 유형 2는 일정분량의 문자열 필기후에 인식을 수행하는 방법으로 사용자는 아무런 제약없이 일정분량의 필기를 진행할 수 있으나 필기자는 필기후 문자열이 인식되는 동안 기다려야 하는 불편이 있다. 유형 3의 방법은 필기와 인식을 병행하므로 필기자는 인식기에 구애받지 않고 필기를 진행할 수 있다. 이러한 방법에서는 복수개의 프로세스가 병행적으로 수행되어야 하기 때문에 신속한 인식시간을 갖는 인식방법을 요구한다. 이 방법에서는 연속적으로 입력되는 필기데이터 중에서 글자분할을 위한 방법론은 필기 유형에 따라 다양한 방법이 적용될 수 있다.

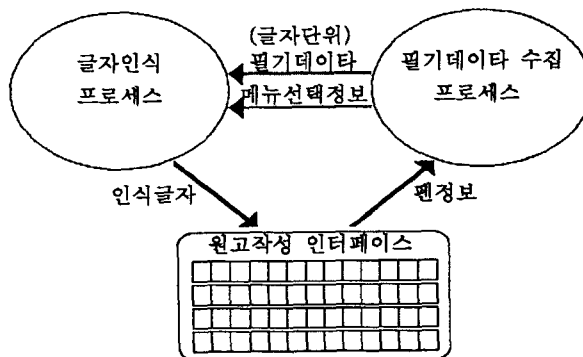


그림 1. 원고작성기의 전체적인 구성도

본 논문에서는 설명하고자 하는 원고작성기는 위에서 열거한 필기형태 중에서 격자내

자소간 홀림을 허용하고, 인식기의 수행되는 시점은 필기동작과 병행되는 구조를 갖도록 구성하였다. 입력된 필기 데이터로부터 글자 분할은 시간적인 정보와 격자의 공간적인 정보를 이용하여 실행하였고, 인식은 분할된 글자단위로 동작한다. 전체적인 시스템 구성도는 그림 1과 같다. 전체적인 시스템은 글자를 인식하는 프로세스, 태블릿으로부터 얻어진 필기데이터와 메뉴선택정보를 수집하여 글자 단위로 글자인식 프로세스로 전송하는 프로세스로 구성된다. 인식되어진 글자는 원고작성 인터페이스에 되돌려져 디스플레이된다.

III. 입력글자의 인식

1. 스트링 생성

입력문자는 인식에 영향을 미치는 필기시의 떨림과 빠침을 최소화하기 위하여 전처리 과정을 거친 후에 등간격의 선분들로 나뉘어진다. 이러한 방식으로 분할된 선분들은 선분의 특성에 따라 가상 선분, 모서리 선분, 일반 선분의 3가지 유형으로 분류된다. 스트링 표현을 위하여 사용되는 알파벳(alphabet)은 이러한 선분의 유형 분류와 선분들의 방향값에 따라서 정의되고, 입력문자는 정의된 알파벳의 기호들을 사용하여 스트링으로 변환된다. 전체적인 스트링 정합과정은 그림 2와 같다.

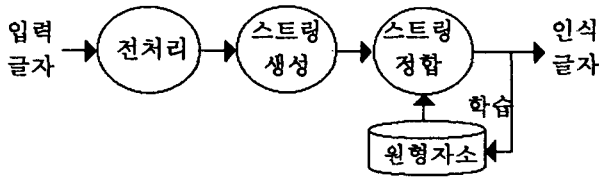


그림 2. 스트링 정합과정

인식시스템의 입력장치로 사용되는 tablet은 필기자의 필기를 이루는 점들의 x,y좌표값과 사용자의 pen-up과 pen-down 정보를 제공한다. 1개의 획은 pen-down과 pen-up사이의 좌표값들에 의해서 구성된다. 일반적으로 입력된 필기문자는 tablet과 전자펜이 갖는 재질에 관련된 떨림과 빠침을 포함하고 있다. 본 논문에서는 이러한 현상들이 인식에 미치는 영향을 최소화하기 위한 인식에 앞선 전처리과정을 갖는다. 이러한 전처리는 평활화(smoothing), 공간필터링(spatial filtering)과 정규화(normalization)과정들로 구성된다. 평활화는 이웃하는 점들과의 평균화를 통하여 필기시의 떨림을 흡수한다[8,14]. 공간필터링은 필기자의 불규칙한 필기속도에 기인하여 연이은 점들의 간격이 필요없이 조밀하거나 동떨어진 경우를 고르게 조정한다. 정규화는 입력되는 문자의 크기를 특정한 사각형(100x120)안으로 입력문자의 높이와 폭을 대응시킴으로써 입력문자로 부터 생성되는 스트링이 문자의 크기에 무관하도록 하였다.

이러한 전처리 과정을 거친 입력문자들은 등간격의 선분들로 나뉘어지며, 획과 획을 연결하는 가상 획도 등간격의 선분들로 나뉘어 표현된다. 나뉘어진 선분들은 다음과 같은 3가지 유형으로 분류한다.

- 가상 선분 : 가상획으로 부터 생성된 선분들
- 모서리 선분 : 자신이 갖는 방향성과 시간적으로 바로 다음인 선분의 방향성의 차이가 급격한 차이를 보이는 선분들
- 일반 선분 : 가상 선분과 모서리 선분에 해당하지 않는 선분들

가상선분은 가상 획의 등간격 분할의 결과로 생성된 선분들로서, 가상획은 입력된 필기데이터 중에서 획과 획을 잇기 위하여 펜이 pen-up하여 다음 획의 pen-down까지의 이동으로 발생하는 펜의 동작을 의미한다.

모서리 선분을 검출하기 위해서는 일반적인 모서리(corner) 검출 방법이 적용될 수 있으며, 일반적으로 영상 정합의 형태분석시 모서리는 매우 유용한 특징으로 알려져 있다 [11,12]. 3장에서 기술할 스트링 정합방법에서는 동일한 유형의 선분사이에서만 정합이 일어나도록 제한함으로써 모서리 특징이 스트링 정합의 복잡도를 효과적으로 줄일 수 있도록 하였다. 본 논문에서는 모서리 선분의 검출을 위하여 Roselfeld-Johnston 알고리즘을 적용하였다[12]. 이 방법은 디지털 곡선상에서 의미있는 곡률의 최대치를 발견하기 위하여, 가변적 크기의 벡터를 생성하고 벡터내적 계산에 의해 구하여진 cosine 값들의 열에서 국소적인 최소값이나 최대값을 검출하여 모서리로 정의하는 방법이다.

입력문자와 원형자소(prototype grapheme)는 방향값에 따라 정의한 기호들로 구성된 스트링으로 표현되어진다. 이러한 기호의 배정은 생성된 선분들마다 1개의 기호씩 이루어지며, 선분의 유형이 다르면 동일한 방향값일지라도 다른 기호들을 사용하여 선분의 유형을 구별할 수 있도록 하였다. 선분의 3가지 분류 유형 중에서 일반 선분들의 방향성을 표현하기 위한 기호들의 집합 V_{NL} 은 다음과 같으며, 각 기호가 의미하는 방향성은 그림 3과 같이 정의된다.

$$V_{NL} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$$

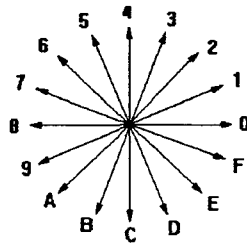


그림 3. 16방향 코드

가상 선분들의 방향성 표현을 위한 기호들의 집합 V_{VL} 은 다음과 같으며, 각각의 기호가 의미하는 방향값은 각 기호에서 '-'를 제거한 V_{NL} 의 기호와 동일하다.

$$V_{VL} = \{\bar{0}, \bar{1}, \bar{2}, \bar{3}, \bar{4}, \bar{5}, \bar{6}, \bar{7}, \bar{8}, \bar{9}, \bar{A}, \bar{B}, \bar{C}, \bar{D}, \bar{E}, \bar{F}\}$$

모서리 선분들의 방향성 표현을 위한 기호들의 집합 V_{BL} 은 다음과 같으며 각각의 기호가 의미하는 방향값은 각 기호에서 '^'을 제거한 V_{NL} 의 기호와 동일하다.

$$V_{BL} = \{\hat{0}, \hat{1}, \hat{2}, \hat{3}, \hat{4}, \hat{5}, \hat{6}, \hat{7}, \hat{8}, \hat{9}, \hat{A}, \hat{B}, \hat{C}, \hat{D}, \hat{E}, \hat{F}\}$$

그러므로 어떤 문자의 스트링표현을 위하여 정의된 알파벳(alphabet) V 은 다음과 같다. 기호 ' Δ '은 입력문자의 끝을 의미하기 위해 사용한다.

$$V = V_{NL} \cup V_{VL} \cup V_{BL} \cup \{\Delta\}$$

2. 스트링 정합 알고리즘

III.1 절의 스트링 생성 방법으로 생성된 두 스트링이 A 와 B 라고 할 때, 두 스트링의 비교를 위해서 다음과 같은 스트링상의 편집 연산들이 정의 되어진다.

- 교환(change) : $a \rightarrow b$
- 삽입(insert) : $\lambda \rightarrow b$
- 삭제(delete) : $a \rightarrow \lambda$

여기서 $a \in A, b \in B$ 이며 λ 은 공백스트링(null string)을 나타낸다. 이러한 3가지 스트링 편집 연산은 비교되어지는 스트링상의 기호들의 유형에 따라 적용여부가 결정되며, 이러한 편집연산자의 적용조건은 표 1과 같다. 비교순서함수는 편집연산에 따라 다음에 비교 되어질 스트링상의 기호의 위치를 정하며 표 2와 같이 정의된다.

표 1. 편집연산자 적용조건

연 산	적 용 조 건
교 환	$A(i) \in V_{NL}, B(j) \in V_{NL}$ or
	$A(i) \in V_{VL}, B(j) \in V_{VL}$ or
	$A(i) \in V_{BL}, B(j) \in V_{BL}$
삽 입	$A(i) \in V_{NL}, B(j) \notin V_{NL}$
삭 제	$A(i) \notin V_{NL}, B(j) \in V_{NL}$

표 2. 비교순서함수

연 산	적 용 조 건
교 환	$(i, j) \rightarrow (i+1, j+1)$
삽 입	$(i, j) \rightarrow (i+1, j)$
삭 제	$(i, j) \rightarrow (i, j+1)$

예를들어 입력자소 A 와 원형자소 B 의 스트링 표현 예는 그림4와 같다. 이 두 스트링 사이의 정합과정을 편집연산, 다음에 비교되는 스트링의 기호순서를 중심으로 나타 내면 표 3과 같다.



A = 1199A[^]C00A

(a) 입력자소의 스트링 A



B = 111888B[^]BFFA

(b) 원형자소의 스트링 B

그림 4. 입력자소와 원형자소 스트링

표 3. 스트링 A와 B의 정합과정중의 편집연산과 소요비용

비교 횟수	정합되는 스트링상의 기호 (입력 ↔ 원형)	편집연산자	다음비교순서
1	1 ↔ 1	교환	(0,0) → (1,1)
2	1 ↔ 1	교환	(1,1) → (2,2)
3	9 ↔ 1	삭제	(2,2) → (2,3)
4	9 ↔ 8	교환	(2,3) → (3,4)
5	9 ↔ 8	교환	(3,4) → (4,5)
6	A ↔ 8	삭제	(4,5) → (5,6)
7	A ↔ B	교환	(5,6) → (5,7)
8	A ↔ B [^]	삽입	(5,7) → (6,7)
9	C [^] ↔ B [^]	교환	(6,7) → (7,8)
10	0 ↔ F	교환	(7,8) → (8,9)
11	0 ↔ F	교환	(8,9) → (9,10)
12	Δ ↔ Δ	교환	없음

IV. 실험 결과

본 인식시스템은 SUN Workstation IPC(10 mips)상에서 구현하였고 Tablet은 WACOM SD-422A 모델을 사용하였다. 구현된 원고작성기의 화면은 그림 5와 같다. 프로세스간의 통신은 2개의 pipe를 사용하여 이루어진다. 사용자가 선택할 수 있는 메뉴는 홀림정도의 구분, 학습, 새화면 등으로 이루어져 있으며, 사용자는 학습버튼을 선택하고 초·중·중성의 자소구분과 필기문자에서 자소 구분 영역을 설정하여 즉석에서 학습을 수행할 수 있는 특징을 갖도록 구현되었다.



그림 5. 구현된 원고작성기의 화면

구현된 인식 시스템의 인식을 측정하기 위하여 구성된 실험자료는 50명의 필기자로 부터 얻은 총 21,076 자음 사용하였다. 원형자소 생성을 위한 자소샘플 데이터베이스를 구성하기 위해서 본 인식시스템에서는 10명의 필기자료를 모으고 그중에서 잘못 필기된 문자를 제외한 문자를 수작업으로 초·중·종성의 자소로 분류하여 총 12,722 자소를 얻어 사용했다. 얻어진 인식률은 표 4와 같다.

표 4. 21,076자에 대한 정인식률

1후보 정인식률	3후보 정인식률
88.96 %	91.23 %

V. 결론

본 논문에서는 온라인 홀림체 한글 필기의 원고작성기의 구현을 설명하였다. 온라인 홀림체 한글 필기의 인식은 심한 글자의 변형으로 인하여 인식과정이 복잡하고 신속한 인식시간을 갖기 어려운 문제로 알려져 왔다. 본 논문에서는 이러한 어려움을 해결하는 방법으로서 스트링 정합방법에 기반한 새로운 정합방법을 사용하였으며, 이 방법에서는 스트링 편집과 비교순서함수에 의해 신속한 인식이 가능하다. 그리고 이러한 정합방법을 응용한 원고작성기는 사용자의 편의를 제공하기 위하여 인식 중에도 필기가 가능하도록 구성되었다. 실험결과, 제안하여 구현되어진 원고작성기가 효율적이고 한글 홀림체의 연속적인 필기에 적합함을 보였다. 이러한 원고작성기는 원고작성 요령과 동일한 방식으로 원고를 작성할 수 있으므로 폭넓은 사용이 기대된다. 계속적인 연구가 진행되어야 할 부분으로서 기술어진 한글 필기의 경우에 기술어짐을 흡수할 수 있는 방법을 들 수 있다.

참 고 문 헌

1. 권 오성, 권 영빈, "선분정합에 의한 흘림체 온라인 한글 인식", 인지과학, Vol.3, No.2, pp. 271-289, 1993.
2. 김 태균, 이 은주, "한글에 적합한 획 해석에 의한 연속 필기 한글의 on-line 인식에 관한 연구", 한국 정보과학회 논문지, Vol.15, No.3, pp.171-181, 1988.
3. 박 재성, 김 성신, 김 태균, "변형된 DP 매칭과 구조해석적 방법을 이용한 흘려 쓴 한글 인식", 제 1회 문자인식 워크샵 논문집, pp.181-184, 1993.
4. 성 태진, 방 승양, "문자조합 규칙 학습에 의한 한글 온라인 필기 인식기의 설계", '91년 가을 학술발표논문집, 한국정보과학회, pp.223-226, 1991.
5. 신 봉기, 김 진형, "은닉 마르코프 모델을 이용한 온라인 한글 인식", 제1회 문자인식 워크샵 논문집, pp.189-194, 1993.
6. 이 성환, 박 희선, "한글 인식의 사례 연구(최근5년 동안의 연구결과를 중심으로)", 제1회 문자 인식 워크샵 논문집, pp.3-pp.46, 1993.
7. 전 병환, 구 본석, 김 성훈, 김 재희, "퍼지이론을 응용한 펜 컴퓨터에서의 On-Line 획인식 기법", 91년 춘계학술발표논문집, 한국퍼지시스템연구회, pp. 168-176, 1991.
8. Sing-Tze Bow, "Pattern Recognition and Image Preprocessing", Marcel Dekker, New York, 1992.
9. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms", The MIT Press, Cambridge, 1992.
10. Shimon Edelman, et al., "Reading Cursive Handwriting by Alignment of Letter Proto-types", International Journal of Computer Vision, Vol.5, No.3, pp.303-331, 1990.
11. Colin A. Higgins and David M. Ford, "A New Segmentation Method for Cursive Script Recognition", International Workshop on Frontiers in Handwriting Recognition, Chateau de Bonas, France. Sep., 1991.
12. Hong-Chih Liu and M.D Srinath, "Corner Detection from Chain-code", Pattern Recognition, Vol. 23, No. 1/2, pp.51-68, 1990.
13. C.C. Tappert and H.H. Jeanty, A Study of Several Accuracy-Improvement methods for Hanwriting Recognition Syatem, IBM Research Report RC 15373, Nov. 1990.
14. C.C. Tappert, Ching Y. Suen, and Toru Wakahara, "The State of the Art in On-Line Handwriting Recognition", IEEE Trans. on PAMI, Vol.12, No.8, pp. 787-803, Aug. 1990.
15. R. A. Wagner and M. J. Fischer, "The string to string correction problem", J. ACM 21, PP. 168-173, 1974.
16. Toru Wakahara, Hiroshi Murase, Kazumi Odaka, "On-Line Handwriting Recognition", Proc. of the IEEE, Vol. 80, No.7, pp.1181-1194, Jul. 1992.
17. P. S. P. Wang, Character & Handwriting Recognition, Singapore, World Scientific, 1991.