

---

碩士學位論文

構造的 特徵을 利用한 筆記體  
한글의 8方向 定形化

濟州大學校 大學院

情報工學科



제주대학교 중앙도서관  
JEJU NATIONAL UNIVERSITY LIBRARY

金 榮 敏

1996年 12月

# 構造的 特徵을 利用한 筆記體 한글의 8方向 定形化

指導教授 李 尙 俊

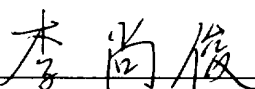

金 榮 敏

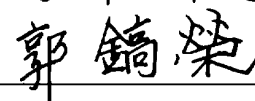
이 論文을 工學 碩士學位 論文으로 提出함

1996年 12月

金榮敏의 工學碩士學位 論文을 認准함

審査委員長  

委 員  

委 員  印

濟州大學校 大學院

1996年 12月

---

**A Study of a 8-direction regularization with  
handwritten Hangul using the structural features**

**Young-Min Kim**

**(Supervised by professor Sang-Joon Lee)**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF MASTER OF**



**DEPARTMENT OF INFORMATION ENGINEERING  
GRADUATE SCHOOL  
CHEJU NATIONAL UNIVERSITY**

**1996. 12.**

# 차 례

Summary .....	I
I. 서 론 .....	1
II. 오프라인 필기체 한글 인식의 특성과 방법론 .....	4
1. 일반적인 문자 인식 기술 .....	4
2. 오프라인 한글 인식 .....	5
3. 입력문자의 전처리 및 특징추출 .....	7
1) 정규화 .....	8
2) 평활화 .....	10
3) 세션화 .....	10
4) 특징추출 .....	15
III. 방향코드 및 퍼지이론을 이용한 특징추출 및 인식 사례 .....	17
1. 필기체 한글의 특성 .....	17
2. 16방향 정형화를 이용한 사례 및 분석 .....	19
3. 8방향 코드 및 퍼지이론을 이용한 사례 및 분석 .....	20
IV. 제안하는 방법 .....	23
1. 특징 추출 및 획 추출을 위한 알고리즘 .....	23
2. 자소가 갖는 구조적 특징을 이용한 추출 획의 8방향 정형화 .....	26
V. 실험결과 및 분석 .....	31
1. 인식 시스템의 구현 .....	31
1) 시스템의 구성 .....	31
2) 실험 자료의 구성 .....	33
2. 정형화 결과 .....	34
3. 평가 및 분석 .....	39
V. 결론 .....	42
참고문헌 .....	43
감사의 글 .....	46

## Summary

In character recognition, preprocessing, extracting feature points and recognition method is generally required.

Especially through the preprocessing, If we can get the unique feature points of each letter, the recognition part will be processed more successfully.

So, we studied a method of preprocessing of the input letter for the recognition of off-line handwritten Hangul. This paper proposed a method that makes the curved point-series data into a 8-directional-lined generalized form using a structural features of Hangul letters.

We also used the informations of location of each input data and fuzzy set theory for 8-direction regularization.

Upper 422 Hangul which used most frequently are tested by the proposed preprocessing method.

The experimental results show that the proposed preprocessing method can extract structural informations of Hangul by the 8-directional-lined form and the output represented the intrinsic features successfully.

# I. 서론

글자인식(Character Recognition)이란 인공지능의 한 분야인 패턴인식의 일 부분으로서, 전자평판(Tablet)과 전자펜(Pen mouse)의 등장으로부터 시작하게 되었다. 전자평판과 전자 펜을 이용함으로써 마치 종이 위에 연필로 필기하는 것처럼 자연스러운 자료 입력을 가능하게 하기 때문에 많은 관심이 집중되어왔다. 최근에는 스캐너의 보급과 함께 스캐너를 통해 이미지로서 입력받은 문서를 인식하여 텍스트로 변환하는 기술까지 등장함으로써 문자인식기술이 널리 상용화되기 시작하였다. 더군다나 데이터베이스 및 네트워크의 발달과 함께 자료공유를 위해서도 기존 문서의 디지털 데이터로의 고속 입력이 가능해진다면 그보다 인간에게 편의를 제고하는 서비스도 없을 것이다. 특히 오프라인 인식 기술은 지리정보시스템(Geographic Information System)을 위한 지도 및 도면의 인식, 우편물의 수신주소 인식, 자동차 번호 판별을 위한 인식 등에도 이용되기 시작하였다.

현재까지 수많은 문자인식 기술들이 연구되어왔으나(권, 1994)(이, 1993a)(이, 1991)(홍, 1991)(심, 1993) 아직까지는 입력할 때 시간 및 방향정보를 이용하는 온라인 문자인식 기술이 주류를 이루고있다. 오프라인 인식 기술도 상당한 발전을 거듭한 끝에 인쇄체 한글인식에 머물던 것이, 다양한 필기체 인식 방법에 대한 논문들이 등장하고 있으며, 고속 무제한 필기체 인식방법(김, 1992b)(이, 1993b)까지 시도되고 있다. 또한 인쇄체도 다양한 글자체와 크기를 갖는 문서(권, 1991), 더 나아가서는 다중 언어를 갖는 문서의 인식까지 연구되고 있다.

그러나, 영어 및 숫자, 한자 등 외국 문자들인 경우는 오래 전부터 연구를 시작하여 거의 완벽한 문자 인식률을 보이고 있으나, 한글의 경우는 1970년대 초반부터 연구되기 시작하여 그 역사가 짧아 아직은 더욱 연구되어야 한다. 필기체인 경우는 사람의 필기특성 및 흘림의 정도에 따라 같은 글자일지라도 그 구조와 모양이 다양하여 인식방법 또한 복잡해지고, 소요시간은 길어지게 된다. 또한 사람의 자연스러운 필기를 받아들이기 위해서는 자소간 및 글자간 흘림을 허용하면서도, 정확한 글자 및 단어추

출을 거쳐, 인식을 해야 함으로 더욱 어려워질 수밖에 없다. 스캐너를 통하여 입력되는 이미지 문서인 경우는, 입력 이미지로부터, 문자영역을 추출하고, 다음에 문자열을, 그리고 글자를 추출하여 인식하는 많은 연구(김,1988)(김,1993)(김,1992a)(김,1992b) 등이 시도되고 있다.

문자인식에 있어서 가장 중요한 부분은 특징추출이라고 할 수 있겠다. 입력 글자의 고유 특징을 유지하면서도, 처리해야할 데이터량을 효과적으로 줄이는 방법은 계속적으로 연구되어야 할 것이다. 이를 위해 많이 연구되고 있는 부분이, 획 추출에 의한 획 정합방법이다. 특히, 이를 위해 체인코드를 이용한 논문들이 등장하고 있다. 글자인식에 16방향 체인코드로써 글자데이터를 표현하고 이를 인식에 사용하기도 하고, (권,1994) 퍼지이론을 이용한 8방향 코드를 사용한 방법도 연구되었다.(전,1994)(김,1995)(심,1992)(홍,1991)

본 논문에서도 획 추출에 의한 인식방법의 하나로서, 필기문자의 다양한 모양보다는 인쇄체 문자의 정형화된 문자가 인식속도 및 인식률이 높음을 감안하여, 필기문자를 8방향의 획으로 구성되는 정형화된 패턴으로 변형하는 방법을 연구하였다. 특히, 한글은 그 구조가 획의 이차원 조합으로 구성되어 있기 때문에 입력문자가 갖는 획의 구조적 특징을 반영하는 정형화된 패턴을 생성할 수 있다면 효과적인 인식데이터로서 역할을 수행할 수 있을 것이다. 이를 위해 한글 사용빈도수 422자에 대한 정자 필기를 실험 데이터로 사용하였다.

이러한 정형화된 패턴을 얻는 여러 방법들이 있다. 이들 중 김(1995)와 전(1994)가 퍼지이론을 이용한 8방향 체인 코드를 이용하였지만, 전(1994)는 획자체의 형태와, 획간의 위치관계를 파악하기 위해 퍼지이론을 사용한 한자의 인식을 연구하였으며, 획의 8방향을 위해서는 일률적인 변환방법을 사용하였다. 또한 김(1995)도 입력 데이터의 획을 추출하여 8방향으로 변형함에 있어서 임계각이 일률적인 온라인 필기인식을 연구하였으며 8방향후 직선의 형태로 변형되지는 않는다.

본 논문에서는 기존 논문들이 갖는 8방향을 위한 일률적인 기울기 값을, 한글이 갖는 구조적인 특성에 맞추어 다양한 기울기 값을 적용함으로써 더욱 효과적인 8방향 정형화 방법을 시도하였다. 또한, 8방향 정형화를 수행할 경우에 글자의 왜곡이 일어나기

쉬운점을 감안하여, 'ㅇ'자에대한 특징추출 방법은 개선된 방법을 적용함으로, 이를 효과적으로 대처하도록 처리하였다.

한글 필기의 구조적 특징을 이용한 8방향 정형화 방법은 결국, 입력 데이터의 8방향 정형화를 통해 인식에 사용되는 데이터 량을 줄이게되고, 입력 데이터의 흘림 정도에 영향받지 않는 정형화된 데이터를 얻는 점에 있어서 장점이 될 수 있다.

본 논문의 구성은 먼저 I 장에서는 서론을 다루고 있고 II 장에서는 일반적인 오프라인 필기체 인식의 특성과 방법론에 대하여 알아보기로 한다. 일반적인 글자 인식의 방법과 전처리 및 세션화, 특징추출 방법 등을 소개하고 있다. 특히 세션화에는 최근에 발표된 4WPTA 병렬 세션화 방법(원,1995)을 소개한다.

III 장에서는 방향코드 및 퍼지이론을 이용한 특징추출 및 인식 사례 분석해보고, IV 장에서는 본 논문에서 제안하는 방법으로 입력문자의 구조적 특징을 퍼지화 데이터로 사용하여 입력데이터의 8방향 정형화를 수행하는 방법을 살펴보기로 한다. 추출된 획을 8방향으로 정형화함에 있어서 어느 정도 신뢰성 있는 방향으로 교정하는 데는 퍼지이론이 적합함으로 퍼지이론에 대한 대략적인 이해와 이를 이용한 정형화를 시도해 보았다.

V 장에서는 실험 결과 및 분석으로, 입력된 필기문자를 8방향 정형화한 데이터를 분석하며 결과에 대한 평가와 문제점에 대해서 살펴본다.

끝으로 VI 장에서는 결론 및 향후 연구과제에 대해서 다루기로 한다.



## II. 오프라인 필기체 한글 인식의 특성과 방법론

### 1. 일반적인 문자 인식 기술

문자인식은 Fig. 1과같이 크게 온라인(on-line)과 오프라인(off-line) 혹은 OCR(Optical Character Recognition) 문자인식으로 나뉘어진다(권,1994). 온라인 문자인식은 마우스를 통해 입력과 동시에 인식이 수행되는 방식을 말한다. 온라인 인식은 글자를 구성하는 획의 순서와 방향 등의 시간적 정보를 함께 가짐으로써 인식이 용이하다. 필기의 정도에 따라 흘려 쓴 필기, 정자 필기의 인식 등으로 구분하기도 한다. 반면 오프라인 인식은 이미 쓰여져있는 글자를 광학 스캐너(Scanner)를 통하여 입력받아서 인식하는 방법을 말한다. 이 방법은 기계에 의해 인쇄된 글자의 인식과 사람에 의해 쓰여진 글자의 경우에 따라서 인쇄체 인식과 필기문자 인식으로 나뉘어진다. 오프라인 인식의 경우 시간정보는 얻을 수 없고, 단지 이미지가 갖는 위상적인 정보만을 가지고 인식을 하므로 온라인 인식보다 더욱 복잡한 시스템 구성을 갖는다(권,1994)(이,1993a).

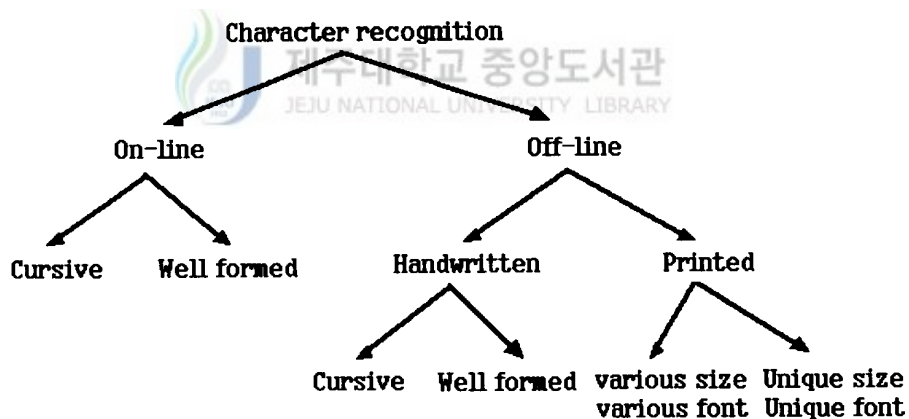


Fig 1. Classification of character recognition

## 2. 오프라인 한글 인식

오프라인 한글 인식은 고품질의 단일 인쇄체 한글부터 시작하여 저 품질의 인쇄체 한글, 다중 활자체 한글을 거쳐 필기체 한글의 인식에 이른다. 오프라인 문자 인식 방법은 크게 원형정합방법, 확률 통계적 방법, 구조적 방법, 신경망을 이용한 방법 등으로 분류될 수 있다(이,1993a).

원형정합 방법은 입력문자영상을 인식 대상이 되는 모든 문자 모델과의 정합을 통하여 유사도나 거리를 구하여 인식하는 방법이다. 그러나 단일 활자체와 같이 변형이 심하지 않은 문자영상은 잘 처리하지만 필기체 문자와 같이 변형이 심한 데이터에는 적합하지 않은 것으로 알려져 왔다. 하지만 비선형 형태 정규화 또는 비선형 형태 정합 방법 등이 개발됨에 따라 원형 정합방법의 성능이 개선되었다.

확률 통계적 방법은 문자영상의 통계적인 특징을 분석하는 방법으로 입력문자 영상으로부터 특징벡터를 추출하는 방법과 대상패턴에 대한 학습과정을 통해 확률 값으로 계산하여 이용하는 은닉마르코프 모델 방법 등이 있다. 그러나 한글과 같이 유사한 글자가 많고 복잡한 문자 집합에 대해서는 이용하기가 쉽지 않다는 단점이 있다.

구조적 방법은 문자를 구성하는 기본요소와 그들간의 연관성을 추출하여 인식하는 방법이다. 복잡한 구조를 갖는 문자 집합의 인식이나 글자 모양의 변형이 심한 필기체 문자의 인식에 적합하다는 장점을 가지고 있으나 기본요소의 추출작업이 어려운 문제가 있다.

신경망을 이용한 방법은 문자인식을 위하여 단순한 기능을 수행하는 처리기들이 대규모 상호 연결된 병렬 분산된 네트워크를 이용한 방법이다. 학습에 의해 새로운 패턴에 적응할 수 있고 패턴의 국부적인 변형 및 잡영에 유리한 점이 있는 반면, 패턴의 크기가 크거나 인식후보대상이 많은 경우 시간이 길어지고 성능이 저하되는 단점을 가지고 있다.

한글 인식문제에 있어서 다른 문자와 비교하여 특성을 갖는다면 다음과 같다

- ◎ 기본자모가 일정 위치에 존재하여, 그 모음의 형태와 종성의 유무에 따라서 6가지의 형태로 분류되는 구조적 특징이 있다.
- ◎ 한글은 'ㅇ'과 직선획 'ㅣ', '-'의 2차원 조합으로 구성된다.
- ◎ 자소 간, 글자간 흘림이 있는 무제약 필기를 허용할 경우 한글은 무한한 서체의 변형이 발생하여 인식의 어려움이 가중된다.
- ◎ 한글 자모의 조합에 의해 이루어지는 글자의 수는 너무나 방대하여 글자단위의 인식보다는 자모단위의 인식이 효과적이다.  
일반적인 오프라인 한글인식 시스템의 구조는 Fig. 2와 같다.



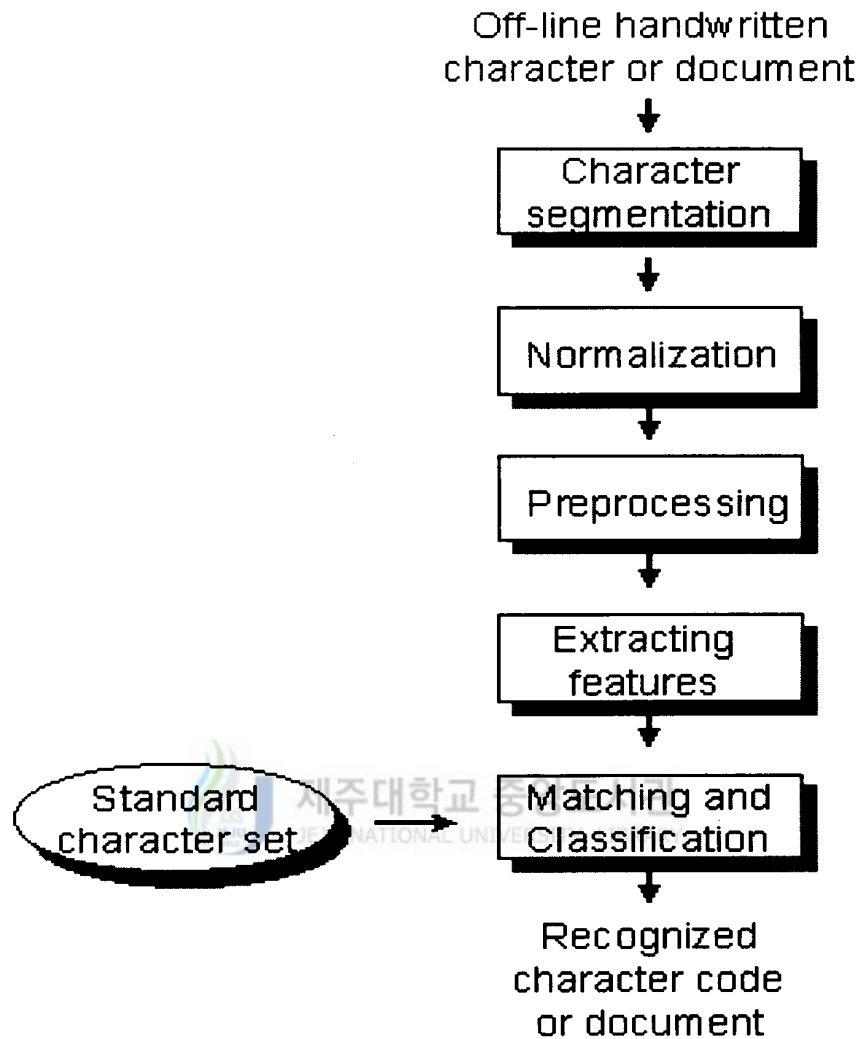


Fig. 2 The structure of character recognition system

### 3. 입력문자의 전처리 및 특징추출

입력문자는 사람의 필기 습관이나, 입력장치의 정확도, 입력할 원본문서의 상태에

따라 인식하기에 불필요한 잡영이 발생할 수가 있다. 또한 정확하고도 쉬운 인식을 위해서도 입력 데이터를 정렬하거나 기울기를 교정하는 등의 정규화 과정, 문서로부터의 문자추출, 문자열로부터의 분자분리의 과정 등도 필요하다. 특히, 세선화 과정은 입력 문자를 인식함에 있어서 처리할 데이터의 양을 줄이고, 효율적인 인식을 위해서 많이 사용되는 전처리 과정의 하나이다.

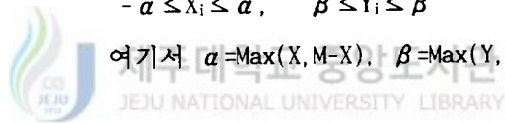
### 1) 정규화

정규화에는 위치정규화(Position Normalization), 크기정규화(Size Normalization), 기울어진 문자 바로잡기(Deskewing), 바닥선 정렬(Base-line drift correction) 등으로 구분된다(이, 1993a).

위치정규화는 위치의 변형에 무관하게 특징을 추출하기 위해서, 문자의 무게중심을 원점으로 하여 위치를 바로잡음으로써 위치를 정규화 할 수 있다. 문자의 무게중심( $X, Y$ )을 문자의 원점,  $M$ 을 문자의 최대크기라 하면, 위치 정규화된 문자데이터는 다음 식과 같은 특성을 갖는다.

$$-a \leq X_i \leq a, \quad \beta \leq Y_i \leq \beta$$

여기서  $a = \text{Max}(X, M-X)$ ,  $\beta = \text{Max}(Y, M-Y)$ 이다.



크기 정규화는 글자의 크기가 크게 입력되거나 작게 입력되어도 올바르게 인식할 수 있도록 글자의 크기를 정규화 할 수 있다. 일반적인 식은 다음과 같다.

$$X_{\ominus} = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}}, \quad Y_{\ominus} = \frac{Y_i - Y_{\min}}{Y_{\max} - Y_{\min}}$$

여기서  $X_{ni}$ ,  $Y_{ni}$ 는 정규화된 데이터이고,  $X_{\max}$ ,  $Y_{\max}$ 는 한 문자에 대한 최대  $X_i$ ,  $Y_i$ 의 값이며,  $X_{\min}$ ,  $Y_{\min}$ 은 문자에 대한 최소  $X_i$ ,  $Y_i$ 의 값이다. 결국  $(X_{\max} - X_{\min})$ 와  $(Y_{\max} - Y_{\min})$ 는 문자의 폭과 높이를 의미한다.

기울어진 문자 바로잡기는 문자의 기울어짐을 정규 할 수 있다. 주로 단어를 대상으로 단어의 중앙을 통과하는 두 수평선을 결정한 다음 두 수평선과 문자가 서로 교차하는 점을 찾는다. 이러한 교차점들을 연결하는 선분의 기울기들을 구하고 이에 대한 평균값만큼 기울기를 변환함으로써 기울기를 교정할 수 있다.

문자의 바닥선 정렬은 문자들의 바닥 선을 맞추으로써 문자의 인식 효율을 높일 수 있다. 먼저 단어내의 Y좌표 값에 대한 모든 국부적 최소 값을 찾은 다음, 이 국부적 최소 값의 두 점들을 잇는 선분을 생성하고, 생성된 선분의 최우도(maximum likelihood)기울기와 관련된 바닥 선을 결정하고 이 기울기에 따라 Y축에 대한 수학적 인 변환을 수행하면 교정된 문자열을 얻을 수 있다.

위 치 정 규 화 → 위 치 정 규 화

(a) Position normalization

크 기 정 규 화 → 크 기 정 규 화



(b) Size normalization

기울어진 문자 → 기울어진 문자

(c) Deskewing

# 바닥선 정렬 ⇒ 바닥선 정렬

(d) Base-line drift correction

## 2) 평활화

평활화는 문자인식에 있어서 중요한 부분인 특징추출시 불필요하게 추출되는 특징점이 없도록 방지하기 위해 잡음제거 및 안정된 데이터 상태를 만들기 위해 입력데이터를 고정해 준다. 다음과 같이 앞뒤의 가중치가 같게 하고 5내지 7개의 점을 평균한 값으로 대체해 주면 된다. (이, 1993a)

$$(1) X_i = (X_{i-2} + X_i + X_{i+1} + X_{i+2}) / 5$$

$$(2) X_i = (-3X_{i-2} + 12X_{i-1} + 17X_i + 12X_{i+1} - 3X_{i+2}) / 35$$

$$(3) X_i = (X_{i-3} + 3X_{i-2} + 6X_{i-1} + 7X_i + 6X_{i+1} + 3X_{i+2} + X_{i+3}) / 27$$



## 3) 세선화

세선화 알고리즘은 인식 대상 영상을 양자화하여 영상을 구성하는 가장 기본 요소인 화소(Pixel)를 다루는 작업으로, 대상 영상의 특징 추출에 무관한 화소를 제거하여 골격선의 두께가 1인 화소로써 그 특징 골격을 형성하는 작업이 된다. 이렇게 세선화된 데이터는 형태분석이 쉽고 데이터량이 적기 때문에 인식속도의 향상을 기대할 수 있게 해준다. 또한 세선화된 표현은 그 패턴에 대한 인간의 개념에 더 가깝기 때문에 보다 간단한 구조분석과 직관적인 인식 알고리즘을 설계할 수 있게 해준다. 이외에도, 중요한 위상적, 기하학적 특성을 유지하면서도 윤곽선의 변형을 제거할 수도 있다. 보다 실제적인 면에서 본다면, 가늘고 긴 패턴의 세선화된 표현은 끝점(end point), 접합점(junction point) 그리고 성분간의 연결과 같은 중요한 특징들의 추출을 보다 용이하

게 해준다. 그러므로 세선화 알고리즘은 인식률을 결정하는 가장 중요한 요인이라 할 수 있겠다. 특히 문자인식에서는 세선화 알고리즘마다 생성되는 골격선의 결과가 서로 다르게 나타날 수 있으므로 인식 대상문자에 적합한 알고리즘으로 수행되어야 한다. 더욱이 문서 인식 시스템(OCR)에서는 여러 종류의 문자와 다양한 활자체가 혼합되어 사용되므로 이러한 문서 인식에 적합한 세선화 알고리즘이 필요하다.

본 논문에서는 입력문자 데이터의 세선화를 위해 4-인접 연결값을 이용한 병렬 세선화 알고리즘(원, 1994)을 사용하였다. 4-인접 연결값을 이용한 병렬 세선화 알고리즘은 다음과 같다.

기존의 세선화 알고리즘들은 대부분이 이미지를 구성하는 모든 화소들을 단순히 0, 1의 상태 값으로 처리하고, 세선화 작업을 위한 지역연산 과정에서 그림 2-3 같은 3×3 windows를 사용하므로 많은 제약 요인이 있었다. 그러나 4-인접 연결값을 이용한 병렬 세선화 알고리즘은 그러한 문제들을 해결 할 수 있게 해준다. 4-인접 연결값이란

P <sub>7</sub>	P <sub>0</sub>	P <sub>1</sub>
P <sub>6</sub>	P <sub>i</sub>	P <sub>2</sub>
P <sub>5</sub>	P <sub>4</sub>	P <sub>3</sub>

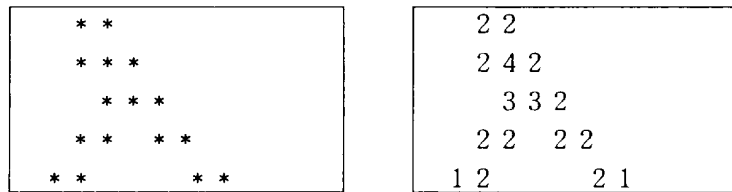
Fig. 3 3×3 Window for thinning

연결관계에 특별한 의미를 부여하기 위해 그림 2-3에서와 같이 한 화소(P<sub>i</sub>)를 중심으로 수평, 수직의 4방향으로 연결된 화소의 합을 말한다. 세선화 작업 시 이미지 데이터를 4-인접 연결값으로 변환하는 과정을 4NCVC(4-Neighbors Connectivity Value Conversion)로 표시한다. 그러므로 모든 화소가 갖는 연결값 4NCV는 0, 1, 2, 3, 4 값 중 하나가 된다.

$$4NCV = \sum_{i=0}^4 P_{2i}$$



다음은 이미지 데이터를 4-인접 연결값으로 변환한 예이다.



(a) Input Image (b) Result of 4NCVC

Fig. 4 Sample of 4NCVC

기존의 세션화 알고리즘들에서 발생하는 문제점은 세션화된 결과가 2화소 두께로 남는 것과 연결성이 끊어지는 것이다. 이 문제를 해결하기 위해서 다음과 같은 3가지 조건을 하나라도 만족하면 제거될 수 없고, 만족치 않는 경우 이를 제거하는 알고리즘을 WPTA(Won's Parallel Thinning Algorithm)라 명한다.

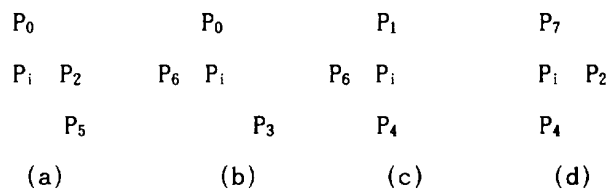
Cond. 1) Connection point

$$\begin{aligned} (P_0 + P_1 + P_2) * (P_3 + P_4 + P_5) * \overline{(P_2 + P_6)} &\geq 1 \\ (P_1 + P_2 + P_3) * (P_5 + P_6 + P_7) * (P_0 + P_4) &\geq 1 \end{aligned}$$

Cond. 2) End point lean

$$B(p) = \sum_{i=0}^7 P_i = 1 \quad B(p): \text{Sum of boundary pixel}$$

Cond. 3) Inclined Y patterns



4 types of inclined Y pattern

위의 4NCVC와 WPTA알고리즘을 결합한 알고리즘을 4WPTA라 명한다.

4NCVC : 화소 데이터를 4-인접 연결값으로 변환

WPTA : 병렬세션화 알고리즘

4WPTA : 4-인접 연결값을 이용한 병렬 세션화 알고리즘

#### 4WPTA Algorithm

단계 1. 이미지 파일을 4NCVC로 변환하고, 연결값으로 변환된 이미지 파일에서 0,1을 갖는 화소를 모두 제거하고, 다시 4NCVC로 변환.

단계 2. 이미지 파일에서 연결값이 0인 화소를 제거한다. (4NCV=0인 화소에 대해 WPTA를 수행)

단계 3. 이미지 파일에서 연결값이 1인 화소를 제거한다. (4NCV=1인 화소에 대해 WPTA를 수행)

단계 4. 이미지 파일에서 연결값이 2인 화소를 제거한다. (4NCV=2 인 화소에 대해 WPTA를 수행)

단계 5. 이미지 파일에서 연결값이 3인 화소를 제거한다. (4NCV=3인 화소에 대해 WPTA를 수행)

단계 6. 단계 2에서 단계5까지 처리 과정에서 제거되는 화소가 있으면 단계 7로 가고, 없으면 종료한다.

단계 7. 단계5까지 처리된 결과를 4NCVC로 변환하고 단계 2로 간다.



#### 4) 특징추출

문자집합에 따라 그 문자만이 가지고 있는 특징이 있다. 이러한 특징에는 시작점과 끝점, 굴곡이 심한 곳, 교차점, 획의 수 등을 예로 들 수 있는데, 문자인식을 위한 특징점이란 그 문자를 대표할 수 있는 특징이 나타나는 지점으로 정의할 수 있다. 따라서 글자인식에 있어서 어떻게 그 문자의 특징점을 찾느냐는 문제는 매우 중요하다. 특징점에 따라서 인식 정도가 현저하게 차이날 수 있기 때문이다. 하지만, 특징점들은 필기자의 습관과 문자의 기울기 및 여러 가지 상황에 따라 Fig. 6과같이 다양하게 나타난다.

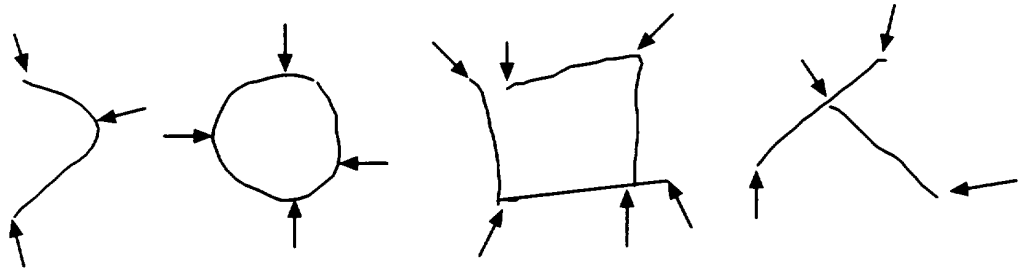


Fig. 6 Feature points from input data

이러한 특징점을 구하는 방법에는 (1) Moving Average를 이용한 방법(SCJH, 1994) (2) 근사방법으로서 최대거리를 이용하는 방법(LGRK, 1995) (3) 최대거리를 이용한 방법(HCMS, 1990) (4) 도함수를 이용한 방법(변, 1994) 등이 있다. 방법(1)은 주로 신호 처리에서 이용되는 방법으로서 여러 입력점에 대해서 평균값에 의해 근사화하여 원래값과의 차를 이용함으로써 특징점을 추출하는 방법이다. 방법(2)는 시작점과 끝점 사이에 존재하는 점들 중에서 시작점과 끝점을 연결하는 직선과 가장 멀리 떨어져 있는 점을 특징점으로 추출하는 방법으로서 간단한 알고리즘에 의해서 구할 수 있다. 방법(3)은 방법(2)와 유사한 방법으로서 최대 거리가 아닌 최소각을 이용한다는 점이 차이가 있으나 대부분의 입력패턴에 대해 찾아내는 특징점 위치가 동일하다. 방법(4)는 입력받은 문자를 구성하는 점들의 좌표값을 미분근사를 이용하여 1차 도함수 값을 구하고, 1차 도함수 값이 0과  $\infty$  그리고  $\pm 1$ 인 부분을 특징점으로 구하는 방법이다.

본 논문에서는 시작점과 끝점을 세션화된 문자 데이터중에서 이웃하는 점의 개수가 1인 점을 시작점과 끝점으로하고, 이웃하는 점의 개수가 3인 점을 접합점, 교차점으로 정의한다. 그리고 이 시작점과 끝점, 그리고 접합점 사이의 굴곡이 심하거나 꺾인 획에 대해서는 방법 2를 변형하여 특징점을 찾아 이용하였다.

```

0 <-- Start-point with 1 neighbor
1
2
Junction point with 3 neighbor -> 2 2 1 <- Most curved point
0 0
1 1 1
End-point with 1 neighbor--> 0 1 <-- End-point with 1 neighbor

```

Fig. 7 Start-point, end-point and curved point



### Ⅲ. 방향코드 및 퍼지이론을 이용한 특징추출 및 인식 사례

#### 1. 필기체 한글의 특성

한글인식에 대한 연구는 지난 20여년 동안 꾸준히 진행되었으나 대부분 인쇄체 한글 인식에 중점을 두고 연구되어 왔으며, 최근 들어서는 펜컴퓨터 개발 열기에 힘입어 필기체 한글의 온라인 인식에 관한 많은 연구가 이루어졌다. 필기 시에 획수, 획순, 각 획에 대한 필기 방향과 획 내에서의 필기 속도 등의 시간적 공간적인 동적 정보를 얻을 수 있는 온라인 인식과는 달리, 오프라인 인식에서는 이러한 정보들을 찾기 힘들거나 전혀 얻을 수 없다. 이러한 정보들을 찾기 위하여 스캐너를 통하여 얻어진 데이터에 대해 윤곽선을 추출하거나, 골격선을 추출하는 등의 부가적인 전처리 과정을 수행해야 하기 때문에 일반적으로 필기체 한글의 오프라인 인식은 필기체 한글의 온라인 인식에 비하여 문제의 난이도가 높다.

지난 수년간 주로 국내의 몇몇 대학을 중심으로 석박사 학위 논문으로써 필기체 한글의 오프라인 인식에 관한 연구가 다양한 방법론들에 의하여 시도되어 왔는데, 이들 방법들은 크게 원형 정합방법, 구조적인 방법, 신경망을 이용한 방법, 통계적인 방법 등으로 분류될 수 있다. 이 중에서 구조적인 방법은 문자의 구성 원리에 입각하여 자소나 자획 등과 같은 문자를 구성하는 기본 요소와 그들간의 연관성을 추출하여 문자를 인식하는 방법으로, 정해진 문법에 따라 음소를 인식한 다음 각 음소의 조합으로 문자를 인식하는 구문론적인 방법과 획을 추출하여 입력 문자의 획과 표준 패턴 획과의 비교를 통해 거리를 계산함으로써 문자를 인식하는 획 정합 방법으로 나눌 수 있다.

획이란 문자를 구성하는 기본 패턴으로서, 특히 한글의 대부분이 획으로 구성된다. 일반적으로 펜으로 글씨를 쓸 때 펜이 지면에 닿은 후부터 펜이 지면에서 떨어질 때까지 펜이 지나간 경로를 의미한다고 할 수 있다. 문자 인식에 있어서 입력 문자 영상으로부터 획을 추출하는 방법은 크게 세선화 단계를 거친 다음 획을 추출하는 방법과 세

선화 단계를 거치지 않고 획을 추출하는 방법으로 분류된다. (김,1993)

세선화 단계를 거친 다음 획을 추출하는 방법은 세선화 단계를 거친 문서 영상에 대해 특징점을 추출하고, 이 특징점들 사이를 연결하는 선분을 매끄러운 연속적 법칙 (Good continuity rule)에 의해 연결하여 획을 추출하는 방법과, 추출한 선분들과 특징점들 사이의 위치 관계를 이용하여 문자를 구성하는 획이나 자소 또는 부패턴을 추출하는 방법으로 나눌 수 있다. 이러한 방법들은 세선화된 결과가 문자영상의 잡영이나 국부적인 정보에 매우 민감하며 처리하는데 많은 시간이 소요된다는 단점이 있지만 세선화 단계를 거침으로써 문자인식과정에 불필요한 화소는 제거하고 최소한의 필요한 화소만을 얻을 수 있음은 물론, 특징점 추출이 가능함으로써 상대적으로 획 추출이 용이하다는 장점이 있다. Fig. 8은 필기 한글이 획의 집합으로 구성됨을 보여준다.

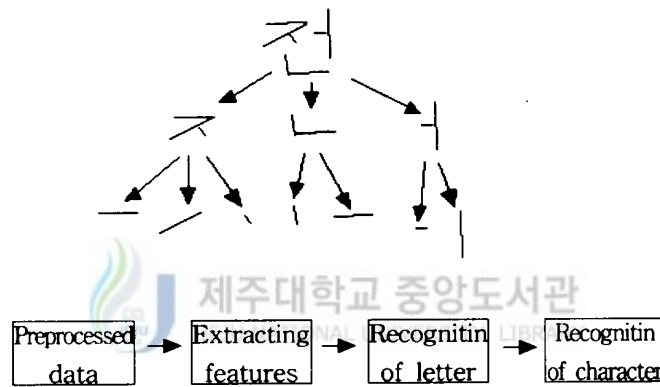


Fig. 8 The structure of character and a flow of recognition

현재 이 획 추출방법을 이용하여 많은 논문들이 연구되어지고 있다. 특히 16방향 및 8방향 Chain-code를 이용한 방법들이 등장하였는데, 김(1995)는 추출된 획의 특징점들을 연결하는 선분을 8방향으로 표현하고 이를 양자화함으로써 얻어진 데이터를 이완정합함으로 온라인 한자인식을 시도하였다. 권(1994)는 추출된 획을 일정간격의 16방향 Chain-code로 변환하여 이에 대한 스트링을 생성하고 이를 원형정합하는 방법으로 온라인 필기체 인식을 시도하였다. 이 외에도 획 추출방법 및 Chain-code를 이용한 필기

체 한글 인식 논문이 많이 연구되어지고 있으나, 본 논문에서는 획추출방법을 이용하여 추출된 획을 다시 8방향으로 정형화된 데이터를 생성함으로써 이를 분류하여 인식하기 쉽도록 연구를 시도해 보았다.

## 2. 16방향 정형화를 이용한 사례 및 분석

권(1994)는 “string 정합 방법을 기반한 온라인 자소 인식”을 위해 입력 데이터를 등간격의 선분으로 나누고, 이를 16방향 체인코드를 사용하여 string을 생성하였다.

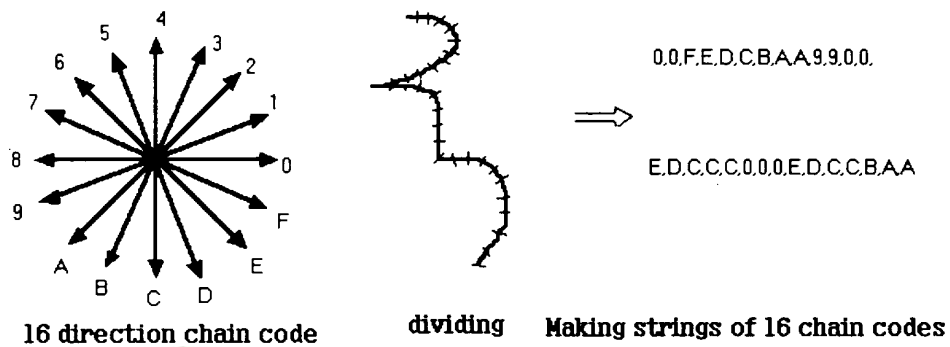


Fig. 9 Creating string using 16-direction chain code

Fig. 9는 입력문자를 등간격으로 분할한 입력데이터를 16방향의 체인코드 값으로 변환하여 인식에 사용할 string을 생성하는 과정을 나타내는 것으로서, 이 연구는 글자를 작은 부분획의 집합으로 표현하여 인식에 사용할 수 있음을 보여주고 있다. 현재 이 16방향의 코드값을 갖는 string은 이미 만들어져 있는 원형집합의 string과 비교됨으로서 삽입, 삭제, 교환 비용이 가장 적게 드는 원형을 입력문자의 인식결과로 선택하게 된다. 이 때 16방향을 사용함으로써 복잡하게 되고, 많은 시간을 소요하게 된다. 그러나 만일 입력문자 고유의 특징을 잃지 않으면서도 이를 8방향, 4방향의 좀더 단순화된 형태로 정형화 할 수 있다면 복잡성과 시간소요의 문제를 줄일 수 있을 것이다. 특히, 한글은 'ㅇ'자를 제외한 모든 자소가 획의 집합으로 구성되기 때문에 충분



히 8방향의 선분만으로도 입력 글자를 표현할 수 있을 것이다. 다만, 'ㅇ'자에 대한 충분한 해결책을 제시해 줄 수 있어야 할 것이다.

### 3. 8방향 코드 및 퍼지이론을 이용한 사례 및 분석

홍(1991)은 각 stroke의 시작점과 끝점을 연결하여 각 특징점에 대한 좌표와 수평방향에대한 각도를 구하여 인식에 데이터로 사용하였다. 이 인식을 위해 각 stroke에 대한 구분을 위해 퍼지화 데이터를 15° 각도로 구분하여 이용하였으며 다음과 같다.

퍼지화 데이터	퍼지값	각도
우측방향 수평선에 극히 가깝다.	0	-7.5° - 7.5°
우측방향 수평선 위쪽에 상당히 가깝다	1	7.5° - 22.5°
위쪽방향 수직선과 우측방향 수평선 사이의 중앙아래에 극히 가깝다.	2	22.5° - 37.5°
.	.	.
우측방향 수평선 아래쪽에 상당히 가깝다	23	-22.5° - -7.5°

이 퍼지화 데이터는 전체 23가지로 구분되어 있으며 이는 16방향이 아닌 23방향 코드 값을 갖는다고 할 수 있으며, 방향값이 많아 질수록 인식에 사용되는 데이터도 많아지며, 계산도 복잡해 질 수밖에 없다.

김(1994)는 온라인 한자 인식을 위해 추출된 등간격의 획을 8방향 코드를 이용하여 중복되는 특징점을 제거함으로써 8방향 양자화 방법을 연구하였다. 그러나 이 방법도 8방향 코드를 반시계방향으로 22.5° 회전 하였을 때, 해당되는 겹치는 특징점을 제거하는 방법으로 8방향의 획으로 구성되는 데이터를 생성하였다.

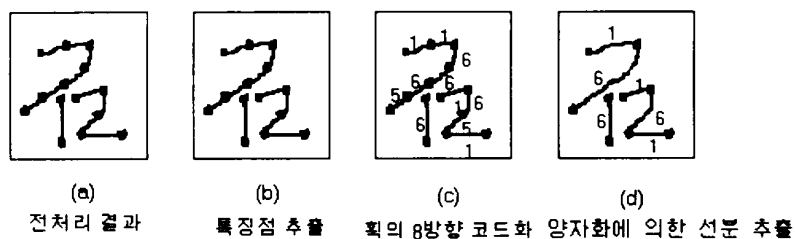


Fig. 10 김(1994)'s extracting 8-direction strokes

전(1994)는 8방향 양자화를 위해 수평선분과 수직선분의 각도를 다르게 하여 정형화 하였다.

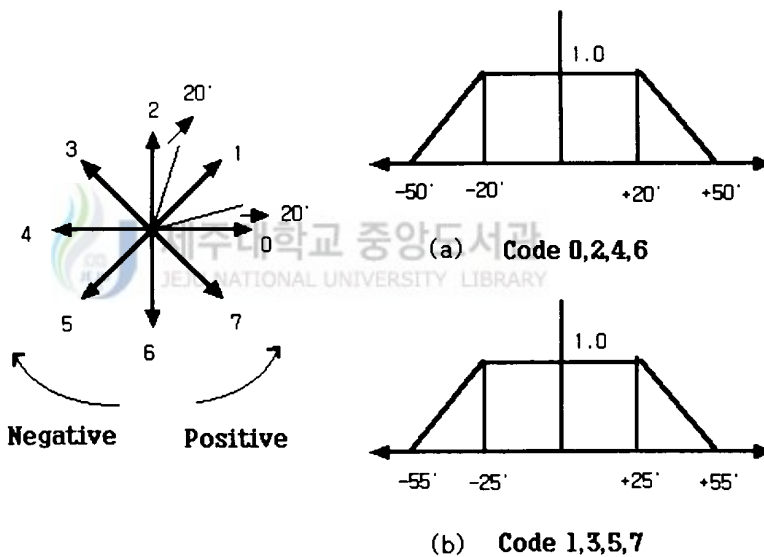


Fig. 11 전(1994)'s membership function for making 8-direction negat

심(1993)은 입력획을 직선성분으로 전처리는 하였지만 8방향 정형화는 하지 않았다.

각 직선성분들은 8방향이 아니므로, 인식에 사용될 때는 복잡도와 시간을 요구한다.  
또한 이 논문에서는 A부분을 연결하여 주지 않으므로 인식에 어려움을 갖게 된다.

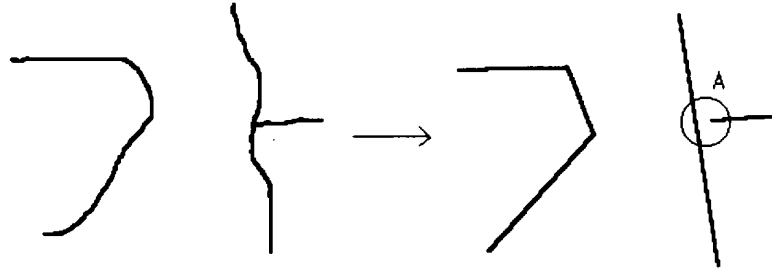


Fig. 12 심(1993)'s making lined form

## IV. 제안하는 방법

III장에서 체인코드 및 퍼지이론을 이용한 연구 사례를 살펴보는 것을 통해 16방향 체인코드 보다는 8방향 체인 코드를 이용할 때, 적절한 정형화만 수행할 수 있다면 빠르고 단순한 인식과정을 수행할 수 있음을 살펴볼 수 있었다. 그리고 여러 가지 8방향 정형화 및 퍼지이론을 이용한 여러 논문들을 살펴보면서 개선되어야 할 점들을 발견할 수 있었다. 이러한 점들을 감안하여 본 논문에서는 입력 문자를 구성하는 자소의 위치정보 및 구조적 정보를 이용한 직선성분의 8방향 정형화를 연구하였다.

이를 위해 먼저 4WPTA 세션화 알고리즘으로 입력된 데이터를 1화소 두께의 점열로 세션화하고, 여기에서 얻어진 8-인접 pixel의 수가 1인 점을 시작점과 끝점으로 하고, pixel 수가 3인 점을 접합점으로 선택하였다. 그리고 거리를 이용한 특징추출을 적용함으로 굴곡점, 꺾인 점을 찾아내고 이 특징점들을 서로 연결함으로써 부분획을 구성하였다. 이 부분획들은 다시 좌표정보 곧 위치정보를 통하여 자음과 모음을 구별해 내고, 이에 대해서 서로 다른 퍼지 데이터를 갖게 함으로 더욱 정확한 인식데이터를 생성하도록 정형화하였다.



### 1. 특징 추출 및 획 추출을 위한 알고리즘

널리 알려진 특징추출 방법에는 4가지가 있다는 것을 앞에서 살펴보았다. 본 논문에서는 특징추출을 위해 2번째 방법인 근사방법으로서 최대거리를 이용하는 방법(LGRK, 1995)을 변형한 방법을 이용하였다. LGRK(1995)는 입력 점열의 시작과 끝점을 찾아낸 후에, 이 두 점을 잇는 직선과 두 점 사이의 점과의 거리가 가장 큰 점을 특징점으로 추출해 낸다. 그러나 본 논문에서는 시작 Pixel로부터 9개의 Pixel로 구성되는 획의 시작과 끝을 잇는 직선의 중점과 획과의 거리가 임계치 이상일 경우, 직선과의 거리가 가장 먼 획의 한 점을 특징점으로 정하고, 하나의 특징점으로부터 다음 특징점까지는 또다시 일정한 간격이상 떨어지도록 임계치를 두었다. Fig. 17을 통해 이를 그

림으로 표현하였다.

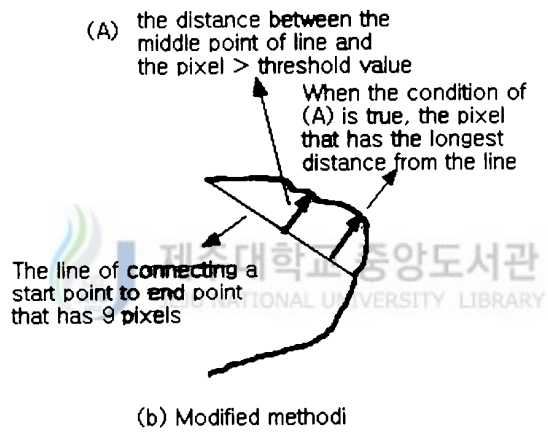
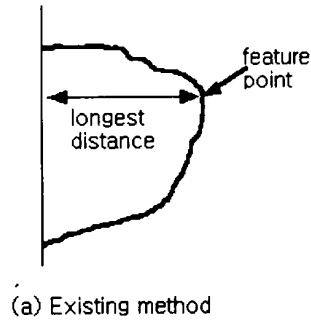
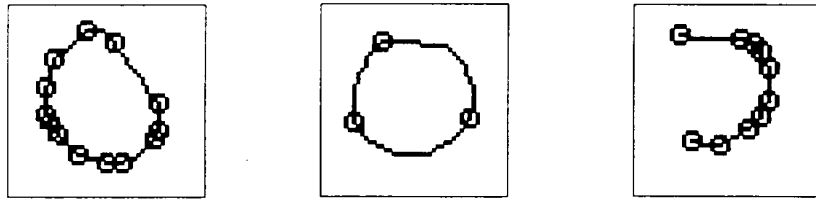


Fig. 13 The method of extracting feature points

여기서 직선과 부분획과의 임계값을 크게하느냐 작게하느냐에 따라 특징점의 수는 달라질 수 있다. Fig. 13은 이에 대한 예이다. Fig. 18의 (a)는 임계값을 작게 한 경우이고, (b)는 임계값을 크게 하였을 때이다.



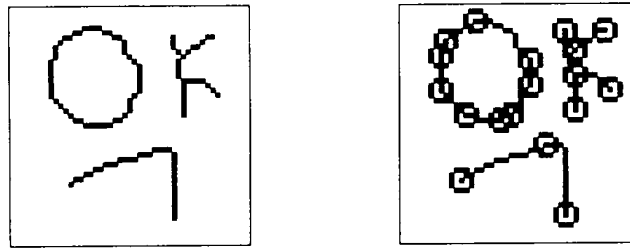
(a) with short distance (b) with long distance (c) with shor distance

Fig. 14 The result of extracting feature points with other distance

한글은 그 구조상 'ㅇ'자를 뺀 나머지는 모두 직선으로 변형하여도 인식에는 크게 지장이 없다. 그러나 'ㅇ'자의 경우는 Fig. 18의 (b)처럼 너무 적은 특징값을 갖게 된다면 'ㅇ'으로서 인식되기가 어렵게 된다. 또한 (c)처럼 단 2개의 직선이면 '7'을 표현할 수 있을 것을, 너무 많은 특징값을 갖게 되면 오히려 불필요한 획수만 늘어나게 됨으로 이를 적절히 조절해 주는 것이 필요하다.

이를 위해서는, 'ㅇ'에 대해서는 작은 임계값을 갖게 함으로 많은 특징점을 갖게 하고, 이외의 경우에는 큰 임계값을 갖게 함으로 정형화의 효율을 높였다.

'ㅇ'에 대해서만 임계값을 작게 하기 위해서는, 시작점으로부터 9개의 점을 선택하여 아주 작은 임계값을 적용시킴으로 초기부터 이를 만족하는 경우는 대개가 'ㅇ'의 경우이므로 계속하여 작은 임계값을 유지하고, 초반에 이러한 작은 임계값에 대해서 특징점을 찾지 못하는 경우는 점차 임계값을 크게 해주어야 한다. 그렇게 되면 'ㅇ'의 경우는 많은 특징점을 갖게 되고, 직선의 경우는 적은 특징점을 갖게 된다. Fig. 15는 이에 대한 예이다.



(a) input data (b) result of extracting feature points

Fig. 15 The result of extracting feature points with 'ㅇ', 'ㅈ' and 'ㄱ'

## 2. 자소가 갖는 구조적 특징을 이용한 추출 획의 8방향 정형화

입력문자로부터 추출된 특징점은 x좌표값과 y좌표값을 갖는 배열로 저장된다. 또한 서로 연결된 점열인 경우는 같은 행에 저장함으로써 자소를 구분하였다. 다음의 예는 '영' 자에 대한 특징점이 저장된 모습이다.

[30, 0], [32, 9], [32, 9], [32, 13], [32, 13], [34, 22], [32, 9], [27, 10], [32, 13], [27, 16] => 'ㅋ'

[15, 10], [19, 15], [19, 15], [19, 21], [19, 21], [10, 17], [10, 17], [14, 11] => 'ㅇ'

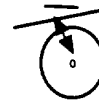
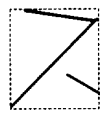
[24, 29], [31, 33], [31, 33], [25, 40], [24, 40], [21, 35], [21, 35], [23, 30] => 'ㅇ'

그러나 서로 연결된 점열만으로 하나의 자소를 구성하게 된다면 한글 필기 입력 시 필기습관에 따라 자소의 모양을 갖추지 못하는 외톨이 획들도 하나의 자소로 인식될 소지가 있게 된다. 김(1995)는 선분의 양끝 점의 거리가  $\rho$  이하이면 이를 이웃하는 점들로 보았지만, 단지 끝점간의 거리만이 아니라 끝점과 다른 획과의 거리로서 근접 획

임을 판단하는 방법을 본 논문에서는 이용하였다. 본 논문에서는 모음이 'ㅣ'나 'ㅡ'가 아닌 외톨이 획은 그 길이가 임계값 이하 이어야 하며, 그 중심이 자소에 임계값 이상으로 가깝거나, 자소의 영역에 임계치 이상 포함되 있을 경우 그 영역에 해당하는 자소의 일부분으로 본다.



(a) 김(1995)'s neighbor stroke



(b) included stroke in a letter's boundary (c) near stroke to a letter

Fig. 16 Extracting a letter from strokes

한글을 구성하는 초성, 중성, 종성의 구조적인 특징을 살펴볼 때, 모음으로 구성되는 중성은 수직획과 수평획으로만 구성된다. 'ㅇ'을 제외한 자음의 경우도, 'ㅅ', 'ㅆ', 'ㅈ' 외에는 사선을 갖지 않는다. 'ㅅ', 'ㅆ', 'ㅈ'의 공통적인 특징은 모두가 접합점, 즉 한 점을 중심으로 주위에 3개의 획으로 연결된 점을 단 1개 갖는다는 것이다. 따라서, 특징점들로 구성된 배열중에서, 자소를 구성하는 하나의 행 내에 접합점이 1개인 경우는 모두가 수직 및 수평획으로 구성되어도 무방할 것이다.

이러한 구조적 특징을 이용하여, 입력 문자를 구성하는 다양한 기울기의 직선획을 적절하게 정형화 하기 위해 퍼지이론을 위한 퍼지화 데이터를 구성하였다.

한글은 다음과 같은 6가지의 구조형식을 갖는다. (도, 1989)

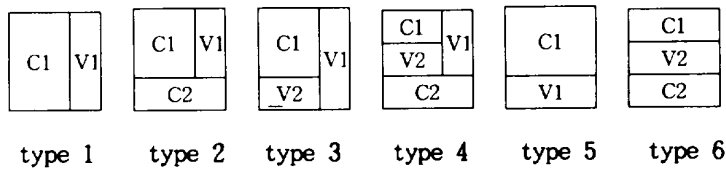


Fig. 17 6 types of Hangul structure



이 중에서 모음에 해당하는 획인 경우는 그 영역의 가로/세로 비율이 어느 한쪽으로 비대칭을 이루게 된다. 즉, 자소를 구성하는 영역의 비율이 어느 한쪽으로 임계값 이상으로 긴 경우는 글자의 모음 부분에 해당한다.

If  $P_x/P_y < \rho_1$  Then  $P_x, P_y$ 의 자소는 수직모음

If  $P_x/P_y > \rho_2$  Then  $P_x, P_y$ 의 자소는 수평모음

(단,  $\rho_1, \rho_2$ 는 모음 추출을 위한 임계값)

모음의 경우는 '/'나 '\'를 갖지 않으므로, 수직, 수평의 직선으로 정형화한다.

수직 모음인 경우는 수직획에 접합점으로 연결된 직선은 수직모음에 수직인 수평획으로 정형화 한다. 또한 수평획에 접합점인 획은 수직획으로 정형화한다.

이와 마찬가지로, 수평모음인 경우는 수평획에 접합점으로 연결된 획은 수직획으로 정형화한다. Fig. 18은 이에 대한 예이다.

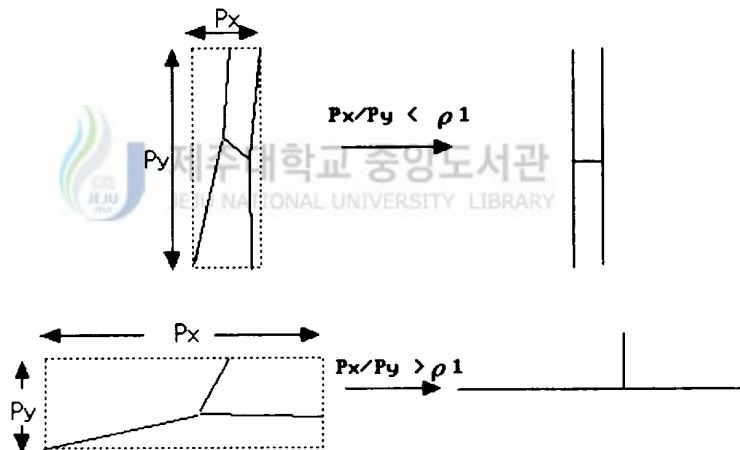


Fig. 18 Extracting of vertical stroke and horizontal stroke

자음의 경우도 'ㅇ'인 경우는 특별한 방법으로 인식에 사용될 데이터를 생성하고, 나머지 자음중에서 'ㅅ', 'ㅆ', 'ㅈ'을 제외한 자소중에서 접합점을 갖지않는 자소, 즉 'ㄱ', 'ㄴ', 'ㄷ', 'ㄹ', 'ㅁ'와 접합점이 두 개 이상인 자소, 'ㅂ', 'ㅍ'는 '/', '\을 갖지 않도록 정형화 한다. 그러나 입력에 사용되는 문자가 필기체인만큼 필기하는 사람의 필기습관에 따라 더욱 다양한 모양과 형태의 문자를 입력하게 되므로, 더욱 많은 접합점을 갖는 자소가 생성될 경우에는 일반적인 임계값을 적용한다.

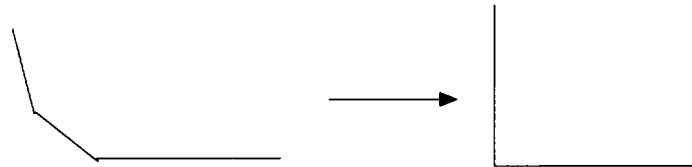


Fig. 19 Regularization of a letter that has no connecting point

퍼지집합은 1995년에 Zedeh 교수에 의해 인간의 주관적 사고나 판단의 애매함을 정량적으로 처리하고자 제안되었으며, 최근에 들어 그 응용이 확대되어 가고 있다.

전체집합 U에서의 퍼지집합 A는

$$\mu_A = U \rightarrow [0, 1]$$


**제주대학교 중앙도서관**  
 JEJU NATIONAL UNIVERSITY LIBRARY

인 멤버십 함수(membership)  $\mu_A$ 에 의해서 특징 지워진 집합으로 정의된다. 이와 같이 퍼지집합은 종래의 Crisp집합에서의 단순한 0,1의 처리에서 벗어나, 인간이 느끼고 생각하는 거의 주관적인 모든 것을 처리해 줄 수 있을 것이다.

추출된 특징량은 필기 변동에 따른 불확실성을 표현하는 퍼지 개념과 대응시켜 퍼지화 데이터로 변환한다.

추출된 특징점들로 구성되는 획의 집합으로부터, 하나의 자소를 구성하는 배열의 횡값을 가지고 수직모음과 수평모음인가를 판단해 내고, 하나의 자소의 영역 내에 접합점의 개수에 따라, 사선 및 역사선 성분을 제거해내도록 퍼지 데이터를 구성한다.

Table 1. 은 본 논문에서 사용되는 퍼지화 데이터이다.

Table 1 Fuzzy set

퍼지화데이터	퍼지 값	각도	자소영역의 x/y 비율	접합점의 갯수	접합점의 위치
수직모음에 해당하며 수직방향의 획에 가깝다.	0	$ \theta  > 45^\circ$	$x/y < \rho_1$		끝점에 아니다.
수직모음에 해당하며 수평방향의 획에 가깝다.	1	$ \theta  < 45^\circ$	$x/y < \rho_1$		끝점이다
수평모음에 해당하며 수직방향의 획에 가깝다.	2	$ \theta  > 45^\circ$	$x/y > \rho_2$		끝점이다
수평모음에 해당하며 수평방향의 획에 가깝다.	3	$ \theta  < 45^\circ$	$x/y > \rho_2$		끝점이 아니다
'ㅌ', 'ㅋ'에 해당하며 수직획에 가깝다.	4	$ \theta  > 67.5^\circ$	$\rho_1 < x/y < \rho_2$	1개	중심부위에 있다
'ㅌ', 'ㅋ'에 해당하며 수평획에 가깝다.	5	$ \theta  < 22.5^\circ$	$\rho_1 < x/y < \rho_2$	1개	끝점이다
'ㅍ', 'ㅊ', 'ㅌ' 해당하며 'ㄹ'획에 가깝다.	6	$22.5^\circ < \theta < 67.5^\circ$	$\rho_1 < x/y < \rho_2$	1개	중심부위에 있다
'ㅍ', 'ㅊ', 'ㅌ'에 해당하며 '\`'획에 가깝다.	7	$-67.5^\circ < \theta < -22.5^\circ$	$\rho_1 < x/y < \rho_2$	1개	끝점이다.
'ㅍ', 'ㅊ', 'ㅌ', 'ㅌ', 'ㅋ'에 해당하지 않으며 수직획에 가깝다.	8	$ \theta  > 45^\circ$	$\rho_1 < x/y < \rho_2$	0개	없다.
'ㅍ', 'ㅊ', 'ㅌ', 'ㅌ', 'ㅋ'에 해당하지 않으며 수평획에 가깝다.	9	$ \theta  < 45^\circ$	$\rho_1 < x/y < \rho_2$	0개	없다.
'ㅍ', 'ㅊ' 밑 기타에 해당하며 수직획에 가깝다.	10	$ \theta  > 67.5^\circ$	$\rho_1 < x/y < \rho_2$	2개 이상	이
'ㅍ', 'ㅊ' 밑 기타에 해당하며 수평획에 가깝다.	11	$ \theta  < 22.5^\circ$	$\rho_1 < x/y < \rho_2$	2개 이상	이
'ㄹ'획에 가깝다.	12	$22.5^\circ < \theta < 67.5^\circ$	$\rho_1 < x/y < \rho_2$	2개 이상	이
'\`'획에 가깝다.	13	$-67.5^\circ < \theta < -22.5^\circ$	$\rho_1 < x/y < \rho_2$	2개 이상	이
'ㅇ'에 가깝다.	14	5개 이상의 방향코드를 갖는다.	$\rho_1 < x/y < \rho_2$	0개	없다.

## V. 실험결과 및 분석

본 장에서는 제안하는 방법의 구현과 결과에 대한 분석 및 평가를 한다. 우선 인식 시스템의 구현에 관하여 설명하고, 구현된 정형화 시스템을 통한 정형화 결과와 이에 대한 평가를 한다.

### 1. 인식 시스템의 구현

#### 1) 시스템의 구성

본 논문을 통하여 제안된 8방향 정형화 시스템은 표 4.1과 같은 환경에서 구현되었다.

Table 2 Configuration of study

Hardware	PC Pentium 90Hz
Operating System	DOS 6.2
GraphicTablet	ACECAT II 5"×5" Graphic Tablet
Programing Language	Turbo C 2.0

5"×5" Graphic Tablet을 이용해 전자 펜으로 사용자 인터페이스 상의 50×50 Pixel 크기의 격자에 글자를 쓰고 이에 대한 Grey Level 0,1의 숫자 값으로 50×50 의 배열에 읽어들이는 데이터를 사용하였다. 입력장치의 불완전함으로 오인식을 유발시키는 잡영이 생기고, 획이 약간 비뚤어지는 현상이 발생하였지만, 평활화 과정과 4WPTA 병렬 세션화 알고리즘에 의해 중심 골격이 1화소 두께로 추출됨으로 이런 문제는 해결될 수 있었다.

주요 처리 과정은 먼저 세션화를 거친 데이터에 대해서 특징점을 추출하고, 특징점들로 구성되는 부분획을 추출한 다음, 각 획에 대한 위치정보 및 기울기등을 퍼지데이터로 만들고 입력되는 획들에 대해서 퍼지를 이용하여 한글의 구조적 특징을 만족하

는 정형화 결과를 얻어내고자 노력하였다.

전체적인 시스템 구성은 Fig. 20와 같다.

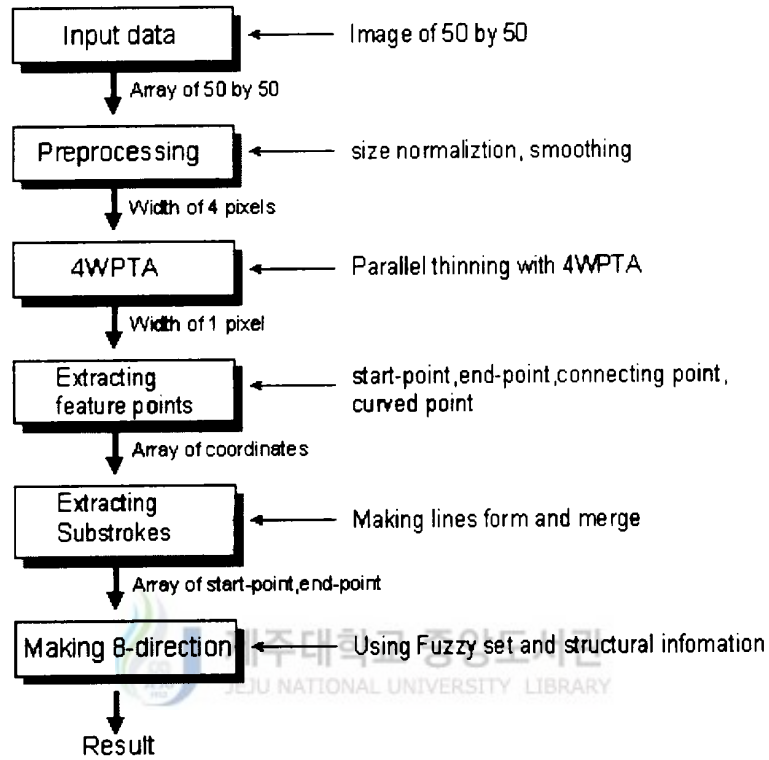


Fig. 20 The structure of 8-direction regularization system

Fig. 21은 실제 입력 및 출력결과를 테스트하는 interface 화면을 보여준다.

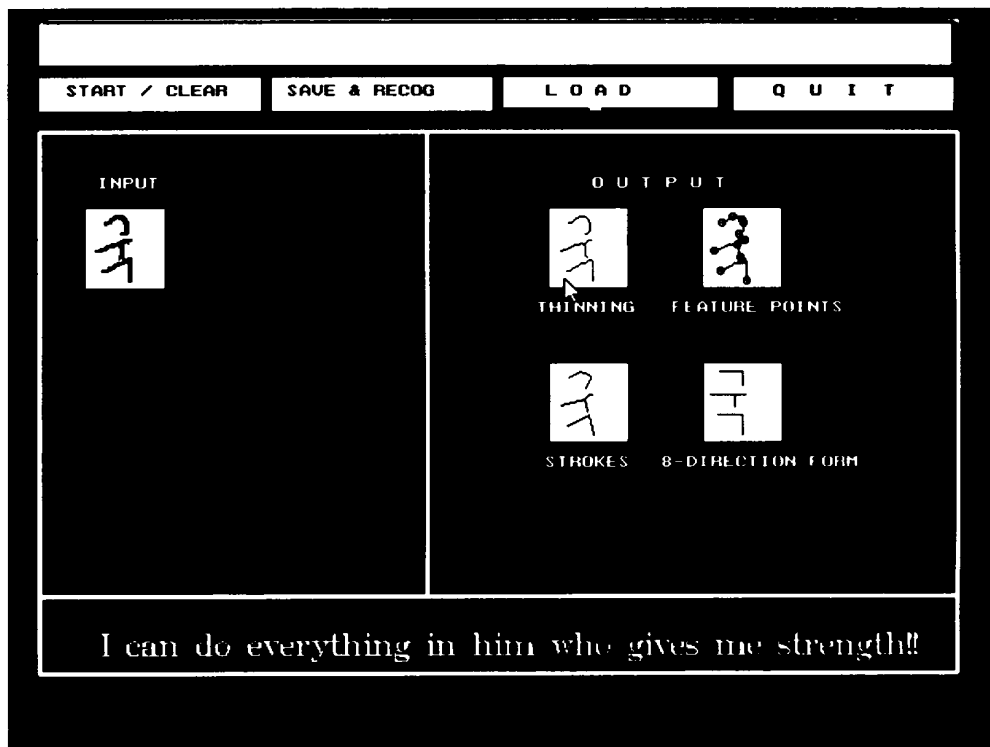


Fig. 21 The interface screen of our system

## 2) 실험 자료의 구성

실험데이터는 한글 사용 빈도수 조사(한글, 1975)에서 제시된 상위 422번째까지의 글자들을 필기하도록 하여 실험하였다. 한글 기계화 연구소에서 제시한 사용 빈도수 상위 422자 안에 포함되는 글자들을 열거하면 다음과 같다.

Table 3 Upper 422 Hangeul character used frequently

이	의	다	는	에	하	을	고	한	가	지	로	서	은	기	대	도	사	있	를	그
인	나	으	어	것	국	들	아	리	정	일	제	수	자	적	시	해	라	과	묘	전
여	주	동	조	보	상	공	성	장	위	개	되	중	화	원	문	만	민	명	우	선
니	회	무	교	계	였	구	신	경	와	할	생	학	당	소	관	러	내	세	실	요
년	미	명	행	비	며	없	히	안	야	마	방	치	발	간	모	개	유	각	단	했
오	데	스	분	말	거	려	남	반	진	바	운	동	결	산	역	근	본	뎠	않	업
양	때	체	강	된	연	작	력	불	용	식	호	명	음	갈	재	건	물	더	월	심
육	또	노	외	복	족	까	저	현	속	던	령	후	등	차	금	포	두	르	활	감
런	직	임	평	설	터	집	쟁	입	점	김	약	람	책	난	배	래	립	예	중	달
든	태	합	침	법	출	합	따	질	양	급	협	및	권	루	격	최	과	포	특	런
친	타	드	새	토	론	받	형	깟	담	독	란	록	청	근	열	처	초	회	느	았
화	키	론	못	슬	많	참	극	철	중	병	트	목	은	환	알	루	석	항	승	혁
괴	앞	추	박	편	계	버	날	변	복	애	준	품	친	크	충	백	레	절	축	변
왔	필	광	디	판	로	탄	될	언	린	음	농	프	규	누	저	별	피	앙	랍	돌
회	완	총	너	늘	령	길	살	락	창	님	순	올	매	건	허	능	접	머	놓	카
망	범	손	옥	잘	존	취	올	코	류	뿐	높	걸	너	링	얼	중	채	써	씨	겨
비	긴	득	밀	쳐	코	억	패	막	맹	습	눈	왕	힘	틈	엔	제	추	색	갓	찾
끝	벌	므	볼	착	량	악	큰	림	삼	엇	위	훈	끼	남	작	황	풍	념	림	송
덕	뎡	뵈	키	뒤	움	찬	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈	뵈
풀	깊	떠	굴	익	줄	쪽	들	렸	갓	돈	엘	졌	찰	뵈	뵈	뵈	뵈	뵈	뵈	뵈

## 2. 정형화 결과

다음은 ‘국’, ‘다’, ‘갓’, ‘엘’, ‘한’에 대한 실제 8방향 정형화를 수행한 결과이다.

(a)는 전자펜으로 입력하여 50×50 크기의 배열로 읽어들이는 값이며, (b)는 이에 대해 4WPTA 세션화 알고리즘을 수행한 결과이다. (c)는 세션화된 입력 데이터로부터 시작점, 끝점, 굴곡점, 접합점을 찾은 결과이고, (d)는 이 특징점들을 서로 연결한 직선획을 보여준다. (e)는 이 직선획들을 다시 8방향으로 정형화한 결과이다.

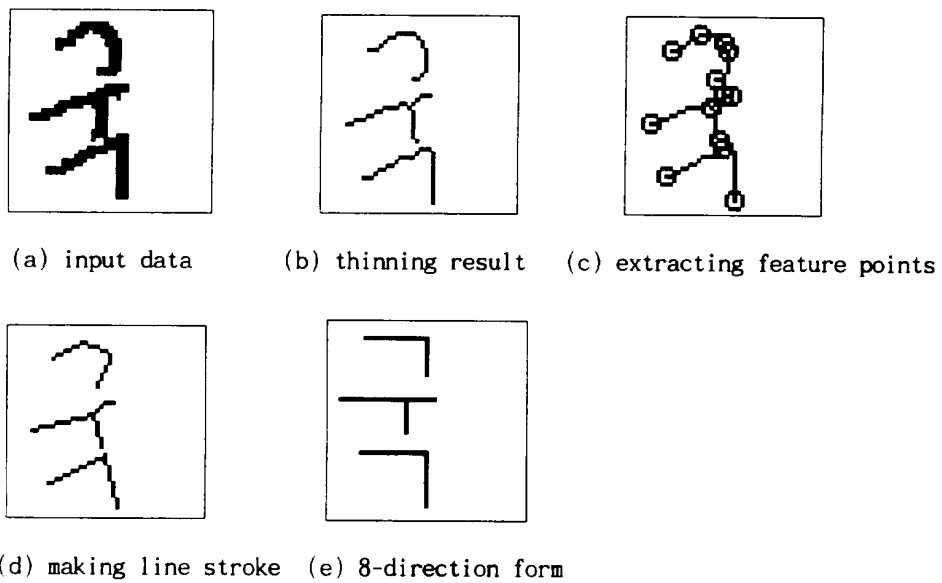


Fig. 22 The result of 8-direction regularization of “국”

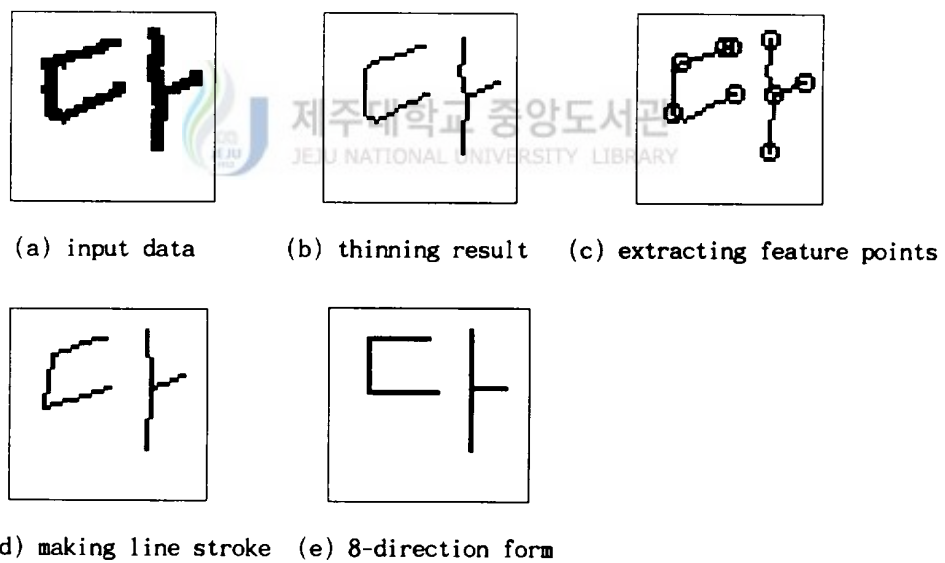
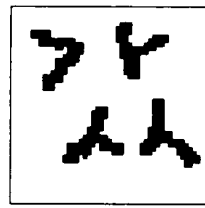
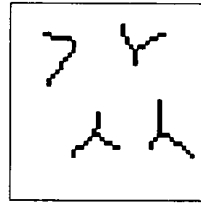


Fig. 23 The result of 8-direction regularization of “다”

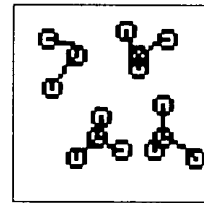




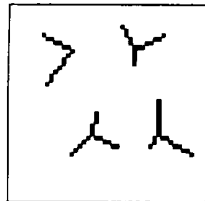
(a) input data



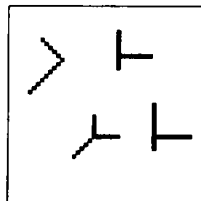
(b) thinning result



(c) extracting feature points



(d) making line stroke



(e) 8-direction form

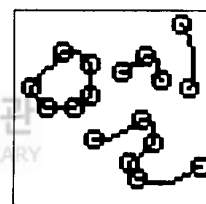
Fig. 24 The result of 8-direction regularization of '갓'



(a) input data



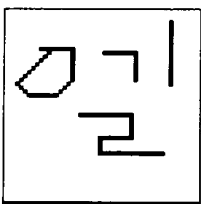
(b) thinning result



(c) extracting feature points



(d) making line stroke



(e) 8-direction form

Fig. 25 The result of 8-direction regularization of '갓'

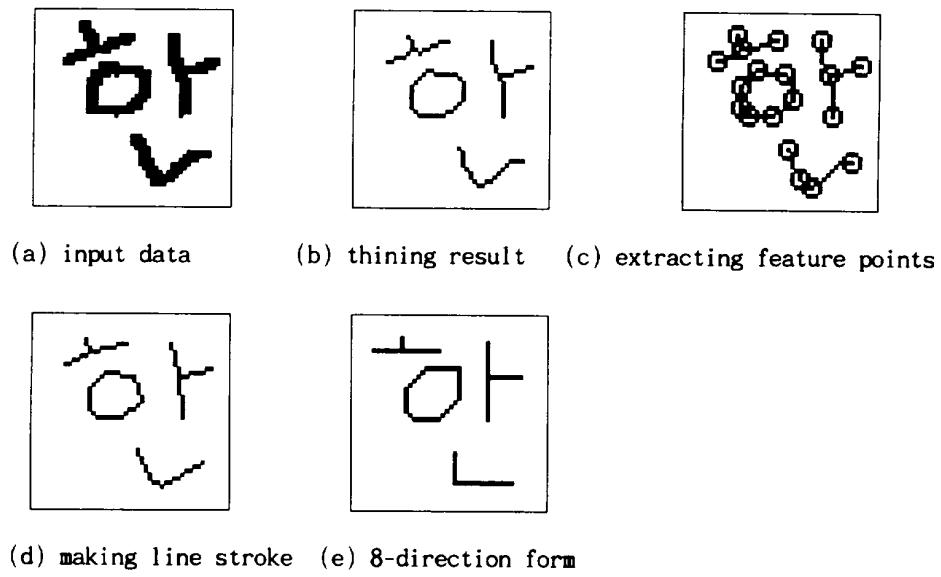
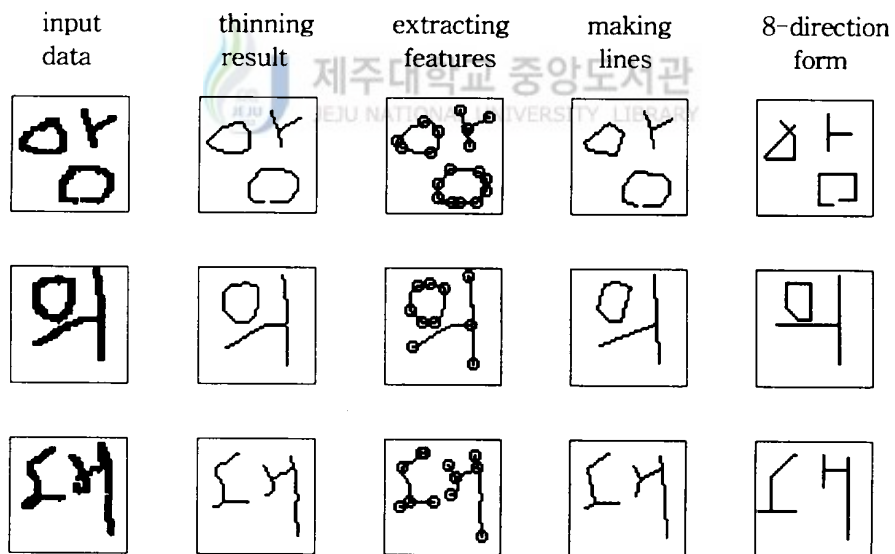


Fig. 26 The result of 8-direction regularization of '한'



input	thinning result	extracting features	making lines	8-direction form
				
				
				
				
				
				
				

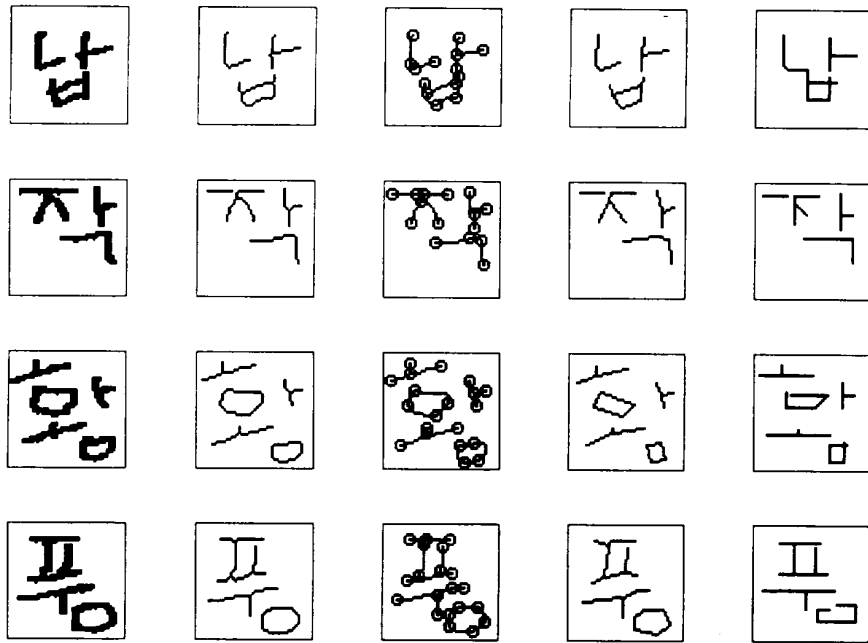


Fig 27. Results of 8-direction regularization

### 3. 평가 및 분석



구조적 특징을 이용한 8방향 정형화 방법은 입력데이터의 고유특징을 유지하면서도 인식에 쉽게 사용될 수 있는 8방향의 단순화된 직선획의 형태로 변형을 효과적으로 수행함을 살펴볼 수 있었다.



(a) 16 directions (b) 8 direction of this paper

Fig 28. Compare 권(1994)'s 16 direction with this paper's 8 direction

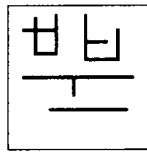
Fig 28. 에서 '국'에 대한 권(1994)의 경우는 총 19개의 획 데이터를 갖지만, 본 논문이 제안하는 정형화된 8방향 데이터는 총 6개의 획 데이터를 갖게 된다. 이러한 본 논문의 정형화 방법은 8방향 정형화 방법은 권(1994)의 16방향 코드를 사용한 입력데이터의 스트링 생성을 보다 간단한 구조로 변형함으로써 더 단순한 정합계산을 요구하게 될 뿐만 아니라, 인식에 사용되는 원형집합의 수도 줄어들게 됨으로 더욱 빠르고 고속 인식을 수행하게 된다.

또한 홍(1991)의 퍼지화 데이터를  $15^\circ$  각도로 구분하여 이용한 것과, 김(1994)의 8방향 코드를 반시계방향으로  $22.5^\circ$  회전하는 방법, 그리고 전(1994)의 8방향 양자화를 위해 수평선분과 수직선분의 각도를  $20^\circ$  와  $25^\circ$  다르게 하여 정형화 한 방법에 구조적인 정보를 이용하여 각 자소의 구조적 특징에 알맞는 8방향 정형화 각도를 퍼지이론으로 적용함으로써 더욱 효과적인 8방향 정형화를 수행 할 수 있다.

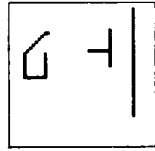
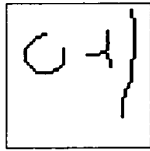
Fig 27. 에서 '국'자의 'ㄱ'나 받침 'ㄱ'의 수평획들은 모두 대각선에 가깝지만, 본 논문이 제안하는 구조적 특징에 따른 기울기 변환을 수행함으로써, 수평획으로의 변환이 가능하게 된다.

그리고 심(1993)의 8방향 정형화를 하지 않았을 때의 처리할 데이터량이 많아지는 것과, 서로 연결된 획이 분리되는 현상을 효과적으로 대처할 수 있었다. '국'자의 'ㄱ'자의 경우도, 'ㄴ'와 'ㄷ'가 서로 분리되기 쉬운 문제를 서로 연결되도록 처리함으로써 글자의 왜곡을 줄일 수 있다.

입력데이터에 대하여 8방향 정형화를 수행하였을 경우, 사람이 보기에는 오히려 더 왜곡된 것 처럼 보일 수는 있다. 하지만, 실제 저장매체에 저장된 입력데이터의 특징 값이 인식에 사용될 경우, 인식성능의 향상을 가져올 수 있을 것이다. 다음은 보기에 왜곡이 심한 글자들의 예이다.



(a) thinning result      (b) 8-direction form



(a) thinning result      (b) 8-direction form

Fig. 29 The side-effect of 8-direction regularization.

Fig. 29의 (a)는 'ㄴ'자의 경우 'ㄴ'자처럼 변형된 모습을 볼 수 있다. 하지만, 실제 저장된 특징값은 2개의 획이 서로 수평으로 연결된 구조를 갖고 있으므로, 인식에 사용되더라도 'ㄴ'으로 인식되지는 않는다. (b)의 경우는 'ㅇ'의 정형화 결과가 전혀 엉뚱한 문자로 오인식 될 수 있는 경우를 보이고 있으나, 실제로는 'ㅇ'의 경우는 다른 어떤 획보다도 많은 특징값을 갖도록 되어있다. 따라서, 많은 특징값을 가지면서도 일정방향으로 그 기울기가 변하는 자소는 'ㅇ'으로 처리할 수 있게 된다. 보기에는 'ㅇ'처럼 보이지 않지만, 실제 저장된 특징값들로 충분히 'ㅇ'으로 인식될 수 있다.

## V. 결론

한글을 포함한 필기문자의 오프라인 인식은 문자를 이루는 획의 홀림정도에 따라 그 인식의 어려움이 더해간다. 그러나 인쇄체인 경우는 인식이 용이하여 많은 연구가 이루어져, 이제는 시중에 판매되고 있는 시스템까지 등장하고 있다. 이러한 점을 감안하여, 필기체 인식을 위해 정형화된 데이터를 생성할 수 있다면 인쇄체 인식의 효과를 얻을 수도 있을 것이다. 이를 위해 본 논문에서는 흘려 쓴 한글 필기를 8방향의 직선획으로 변형하는 8방향 정형화 방법을 연구하였다.

50×50 Pixel 크기의 입력된 문자 영상 데이터에 전처리, 세션화, 특징추출을 거쳐 한글의 구조적 특징을 이용한 8방향 정형화를 적용한 결과 만족할 만한 결과를 얻을 수 있었다. 특히, 16방향이 갖는 처리할 데이터의 과다로 인한 처리속도의 저하문제나, 방향코드를 이용하지 않을 경우의 입력 또는 학습해야할 원형데이터의 양이 많아지는 문제를 해결하는 효과를 가져올 수 있다. 특히, 8방향 정형화는 필기자의 홀림정도나 필기습관에 따라 인식이 달라지는 일반 인식방법의 문제점을 보완할 수 있으리라 기대된다. 그러나 본논문은 아직 자소내 홀림을 허용하는 정자체 필기의 경우에 불과하다. 이보다 더 복잡한 고속, 무제한 필기 인식을 위하여서는 여기에 자소를 분리해내고, 필기습관에 의해 형성되는 불필요한 획을 분별해 낼 수 있는 알고리즘까지 추가되어야 더욱 정확한 인식을 수행할 수 있을 것이다. 또한 본 논문과같이 8방향의 정형화된 형태의 입력데이터를 생성하는 알고리즘을 효과적으로 흡수하여, 8방향 정형화된 입력 데이터를 효과적으로 인식에 사용하는 인식 시스템 개발이 필요하다.

---

## 참고문헌

---

변영철, 1995, “도함수를 이용한 특징추출 및 온라인 문자인식”, 연세대학교 전산과 학과 석사학위 논문

도정인, 1989, “다중활자체 한글문서의 인식을 위한 한글 유형 분류에 관한 연구”, SAITEC, vol. 2, no 2, pp.65-72

한글기계화 연구소, 1975, 한글기계화 연구

Hong-chin Liu and M.D Srinath, 1990, “Corner Detection from Chain-code”, Pattern Recognition, Vol 23, No 1/2, pp.51-68

홍주표, 장희돈, 남궁 재찬, 1991, “퍼지이론을 이용한 온라인 필기체 한글 문자 인식에 관한 연구” 제 4회 신호처리 합동 학술대회 논문집 제 4권 제 1호, pp.141-144



전병환, 김성훈, 김재희, 1994, “퍼지 결정 트리를 이용한 온라인 필기 문자의 계층적 인식” 전자공학회논문지 제 31권 B편 제 3호, pp.132-140

김병래, 박인갑, 서동필, 김에늑, 1992, “CFG방법을 이용한 필기체 한글에서의 자소 추출과 인식에 관한 연구”, 전자공학회논문집 제29권 B편 제9호, pp.8-16

김기철, 김영식, 이성환, 1993, “필기체 한글의 오프라인 인식을 위한 획 정합 방법”, 전자공학회 논문지 제 30권 B편 제 6호, pp.76-85



김기철, 이성환, 1995, “이완 정합을 이용한 획순에 무관하게 흘려 쓴 한자의 온라인 인식”, 한국정보과학회 논문지, 제 22권 제 3호, pp.423-430

김종렬, 정호선, 이우일, 1990, “특징점 추출에 의한 한글 문자 인식 및 전처리용 신경 칩의 설계” 전자공학회논문지 제 27권 제 6호, pp.114-121

김민석, 손한용, 최완수, 김수원, 1992, “자소추출 방법을 이용한 고속 한글인식 시스템의 구현” 전자공학회논문지 제 29권 B편 제 6호, pp.25-31

태균, 1988, “stroke조합에 의한 필기체 한글의 표현과 인식”, 대한전자공학회 논문지, 제 25권 제 1호, pp.18-26

권재욱, 1991, “계층적 신경망을 이용한 다중 크기의 다중 활자체 한글 문서 인식” 석사학위 논문, 한국과학기술원 전자공학과

권오성, 권영빈, 1995, “스트링 정합 방법을 기반한 온라인 자소 인식” 한국정보과학회논문지 제 21권 제 5호, pp.841-850



권오성, 권영빈, 1995, “한글의 자소간 흘림 필기를 허용하는 온라인 인식 시스템” 한국정보과학회논문지 제 22권 제 2호

권오성, 1996, “스트링 정합을 이용한 흘려쓴 한글 필기의 온라인 인식”, 중앙대 전자공학과 박사학위 논문

이희진, 성태진, 방승양, 1991, “신경회로망을 이용한 한글문자 인식기술”, 전자공학회지, 제 18권, 10호, pp.727-738

이진수, 권오준, 방승양, 1996, "개선된 자소 인식 방법을 통한 고인식률 인쇄체 한글 인식" 정보과학회논문지B, 제 23 권 제 8호, pp.841-850

L. O'Gorman and R. Kasturi, "Document Image Analysis", IEEE Computer Society Press, Los Alamios, Califonia

이성환, 1994, "문자인식 이론과 실제", 홍릉과학출판사, pp.41-129

이성환, 1994, "문자인식 이론과 실제" 제 II 권, 홍릉과학출판사, pp.327-352

심동규, 함영국, 박래홍, 1993, "DP 매칭과 퍼지이론을 이용한 홀림체 온라인 한글 인식", 전자공학회논문지 제 30권 B편 제 4호, pp.116- 129

Soo-Chang Pei and Ji-Hwei Horng, 1994, "Cornet Point Detection Using Nest Moving Average" Pattern Recognition, Vol. 27, No.11, pp.1533-1537

원남식, 손윤구, 1995, "4-인접 연결값을 이용한 병렬 세션화 알고리즘", 한국정보과학회지 제 22권 제 7호, pp.1047-1055

## 감사의 글

제게 있어서 기대감, 설렘 그리고 두려움을 주었던 대학원 생활이 이제 다 되어 갑니다. 그 동안 저를 웃을 수 있게도 하고 인상을 찌푸리게도 하던 수많은 사건과 사연들이 있다가 사라져 갔습니다. 그때마다 저의 인생가운데 변함없이 함께 하여 주신 주님께 가장 먼저 감사를 드립니다. 그리고 무엇보다도, 저를 받아주시고 2년 동안 항상 곁에서 지도하시고 격려와 조언, 모든 배려를 아끼지 않으신 이상준 교수님께 진심으로 감사를 드립니다. 또한 항상 관심을 가져주신 김장형 교수님, 안기중 교수님, 곽호영 교수님, 변상용 교수님, 송왕철 교수님께도 감사를 드립니다.

언제나 항상 옆자리에 앉아 같이 공부하던 회국씨, 그리고 가장 어린 저를 옆에서 아껴주시고 여러모로 도와주신 대학원 동기이자 학과선배, 사회 선배인 김대영, 문석환, 양행진, 김정희, 고영호 선배님들께도 감사를 드립니다. 이외에도 항상 볼 때마다 웃게 하는 유경이, 대학원 1년 후배이지만 저보다는 선배들이신 송창용, 강경숙 후배 님들께도 그 동안 저를 위해 주신 모든 진심의 마음들에 감사를 드립니다.

이외에도 대학원 기간동안 알게된 이들과 지난 2년 동안 제게 조금이나마 도움을 주신 모든 이들에게 이 글을 통해 감사를 드립니다.

끝으로, 오늘 이 순간이 있기까지 저를 위해 항상 한없는 사랑과 관심을 주셨으면서도 부족해 하시는 부모님들, 형님, 누나들에게 감사들 드리며, UBF 모든 형제, 자매님들 그리고 박디모데 목사님, 박한나 사모님, 조마가 목사님, 조 사라 사모님과 정신 박약아로 정상적으로 살아갈 수 없는 조카 윤호를 위해 이 논문을 바칩니다.