

이웃 각도 히스토그램 및 변형된 하우스도르프 거리를 이용한 ‘ㄱ’, ‘ㅇ’ 자소 인식

김도현^{*}, 장원두^{**}, 김하영^{***}, 차의영^{****}

요 약

한글 문자 인식에 있어서 ‘ㄱ’과 ‘ㅇ’의 오인식은 전반적인 인식 성능의 저하를 가져오는 요소가 되고 있으나 이에 대한 연구가 미흡한 실정이다. 따라서, 본 논문에서는 ‘ㄱ’과 ‘ㅇ’을 효과적으로 인식하기 위한 새로운 특징 추출 방법을 제안하였다. 제안하는 방법은 변형된 하우스도르프 거리를 이용한 최적의 이웃 반경을 설정하고, 이 반경에 의해 이웃 픽셀과의 각도를 추출하여 두 자소를 구분하는 특징으로 사용하였다. 실험을 통하여 분석한 결과 제안하는 특징 추출 방법은 기존의 방법들보다 적은 특징 개수를 사용하여 효율적으로 패턴을 인식할 수 있었으며 우수한 일반성 및 안정성을 나타내었다.

The Recognition of Grapheme ‘ㄱ’, ‘ㅇ’ Using Neighbor Angle Histogram and Modified Hausdorff Distance

Do-Hyeon Kim^{*}, Won-Du Chang^{**}, Ha-Young Kim^{***}, Eui-Young Cha^{****}

ABSTRACT

The classification error of ‘ㄱ’, ‘ㅇ’ is one of the main causes of incorrect recognition in Korean characters, but there haven’t been enough researches to solve this problem. In this paper, a new feature extraction method from Korean grapheme is proposed to recognize ‘ㄱ’, ‘ㅇ’ effectively. First, we defined an optimal neighbor-distance selection measure using modified Hausdorff distance, which we determined the optimal neighbor-distance by. And we extracted neighbor-angle feature which was used as the effective feature to classify the two graphemes ‘ㄱ’, ‘ㅇ’. Experimental results show that the proposed feature extraction method worked efficiently with the small number of features and could recognize the untrained patterns better than the conventional methods. It proves that the proposed method has a generality and stability for pattern recognition.

Key words: Consonant Classification(자소분류), Neighbor-Angle Histogram(이웃각도 히스토그램), Handwritten Character Recognition(필기체문자인식), Feature Extraction(특징추출)

1. 서 론

현대 사회가 정보화 사회로 진입하고 급변하는 컴퓨터의 발달에 따라 우리의 정보 처리 능력은 급진전하고 있으나, 아직도 컴퓨터 자료 입력의 많은 부분

들은 사람의 손에 의하여 수행되는 실정이다. 현재 대부분의 자료나 정보가 문자의 형태를 지니고 있으므로 문자 인식의 자동화는 정보 사회 진입을 앞당기는 필수적인 단계라고 보인다. 이러한 관점에서 알파벳, 숫자, 한글 등의 문자인식을 위한 시도는 이미

* 교신저자(Corresponding Author) : 김도현, 주소 : 부산시 금정구 장전동 산30번지(609-735), 전화 : 051)510-2878, FAX : 051)515-2208, E-mail : unlimmit@hanmail.net
접수일 : 2004년 6월 24일, 완료일 : 2004년 10월 12일

[†] 준회원, 부산대학교 대학원 컴퓨터공학과 박사과정

^{**} 부산대학교 대학원 컴퓨터공학과
(E-mail : 12cross79@hanmail.net)

^{***} 부산대학교 대학원 컴퓨터공학과
(E-mail : tippy99@hanmail.net)

^{****} 종신회원, 부산대학교 컴퓨터공학과 교수
(E-mail : eycha@pnu.edu)

* 본 논문은 부산대학교 컴퓨터 및 정보통신 연구소의 지원으로 수행되었음.

많은 연구가 진행되었다[1-4].

한글 문자 인식 방법은 크게 입력 영상을 그대로 사용하는 방법과 원 영상으로부터 특징을 추출하여 인식에 사용하는 방법으로 나눌 수 있다[5]. 원 영상을 그대로 인식에 사용할 경우 문자 영상의 자체의 고유한 특징을 얻을 수 있기 때문에 비교적 안정된 인식률을 얻을 수 있다는 장점은 있으나 인식을 위해 필요한 정보가 늘어나므로 다양한 패턴들의 형태를 처리하기 위한 비용이 커지게 되며 결과적으로 전체적인 인식 성능을 떨어뜨리는 단점을 가지고 있다. 입력 영상의 특징을 사용하는 방법은 상대적으로 적은 양의 정보를 이용하므로 다양한 패턴의 변형을 처리할 수 있는 장점을 가지고 있으나 특징을 추출하는 과정에서 발생하는 전처리 비용에 상대적으로 많은 시간을 요구하게 된다.

특히, 한글의 경우 다른 문자들과는 달리 자소 단위로 초성, 중성, 종성이 조합되어 하나의 문자로 완성되는 특징을 가지고 있으므로 이런 자소의 분할과 인식은 문자 인식의 성능을 결정하는데 매우 중요한 역할을 한다. 그 중 자소 인식은 지난 수년간 연구가 활발히 이루어져 많은 성과들이 있었으나, 구분하기 모호한 자소인 'ㄱ'과 'ㅇ'에 대해서는 인식률이 낮은 문제점을 가지고 있으며[6-9] 결과적으로 전체적인 인식률이 낮게 나타나는 현상이 발생하게 된다. 'ㄱ'과 'ㅇ'은 쓰는 사람마다 글자체가 다르고 상황에 따라서도 다양한 형태를 가지므로 이에 대한 적극적인 연구가 필요하지만 기존의 대부분의 연구들이 완성된 글자의 인식에 초점이 맞추어져 있기 때문에 실제 자소 단위 인식에 대한 집중적인 분석과 연구는 매우 미흡한 실정이다. 따라서 본 논문에서는 'ㄱ'과 'ㅇ'의 기하학적인 특성에 초점을 맞추어 이웃 각도 히스토그램 및 변형된 하우스도르프 거리(Hausdorff distance) 방법을 사용한 'ㄱ'과 'ㅇ'의 효율적인 인식 방법을 제안하고자 한다. 제안하는 방법은 먼저 자소 영상을 반복 임계치 적용 방법[10]을 사용하여 이진화를 한 후 세선화(thinning)하고 획을 구성하는 각각의 픽셀을 대상으로 이웃 픽셀과의 각도 성분을 특징으로 추출한다. 이때, 최적화된 이웃 픽셀의 거리를 판단하기 위하여 변형된 하우스도르프 거리의 특성을 기반으로 최적 중심 반경을 설정하였다.

본 논문의 순서는 다음과 같다. 우선 2장에서는 자소 인식을 위한 특징 추출 방법에 대하여 살펴보고

3장에서는 이웃 각도 히스토그램을 이용한 자소 특징 추출 방법과 변형된 하우스도르프 거리를 이용한 최적의 이웃 반경 설정 방법에 대해 설명한다. 4장에서는 제안하는 특징 추출 방법과 기존 특징 추출 방법들과의 비교 실험을 통해서 자소 인식 성능을 분석하고 5장에서 결론 및 향후 연구 방향을 제시한다.

2. 특징 추출 방법 (Feature extraction method)

문자 인식에서 특징 추출의 방법을 선택하는 일은 매우 중요하다. 특징 추출은 원 영상에서 정보를 뽑아내는 단계이기 때문에 특징 추출 방법에 따라 중요한 정보가 추출될 수도 손실될 수도 있다. 뿐만 아니라 복잡한 특징 추출 방법을 사용하거나 너무 많은 양의 특징을 추출하게 되면 전체 인식 시간에 영향을 미쳐 시스템의 성능을 저하시킬 수도 있다.

지금까지 자소 인식에서 사용되어 온 특징 추출 방법들은 크게 전역적 특징 추출 방법, 통계적 특징 추출 방법, 지역, 위상적 특징 추출 방법의 3가지로 분류할 수 있다[6].

2.1 전역적 특징 추출 방법

전역적 특징 추출 방법은 영상 내에 있는 모든 픽셀의 상태를 특징으로 추출하는 것이다. 영상을 FFT, HARR, DCT, Walsh 등의 주파수 변환을 통하여 영상 신호를 특징으로 사용하는 방법이 여기에 포함된다. 이 방법은 잡영에 덜 민감하다는 장점이 있으나, 수행속도가 느리고 문자의 변형에 너무 민감하여, '!', '-', 'o', '/', '\ ' 등의 간단한 획들의 조합으로 이루어지는 한글 자소의 특성상 한글 자소 인식에 적합하지 못하다[7].

2.2 통계적 특징 추출 방법

통계적 특징 추출 방법은 영상의 통계적인 분포를 특징으로 이용하는 방법이다. 통계적인 방법은 전역적인 방법에 비해 특징의 개수가 비교적 적으며 이에 따라 특징 추출의 속도가 빠른 장점을 가진다. 통계적 특징에는 메쉬(mesh) 특징, 투영(projection) 특징, 교차 거리 특징(cross distance) 등이 있다.

가. 메쉬(mesh) 특징

영상의 가로 및 세로 크기를 일정한 블록(block)으로 나누어 각 블록에서의 평균 명암값을 특징으로 이용한다. 영상을 $M \times N$ 으로 정규화하였을 경우 총 $(M \times N)$ 개의 특징이 추출되며, 이런 mesh 특징은 전체적인 획의 명암 분포를 잘 반영할 수 있지만 특징의 개수가 매우 많고, 획의 이동에 민감한 단점을 가지고 있다.

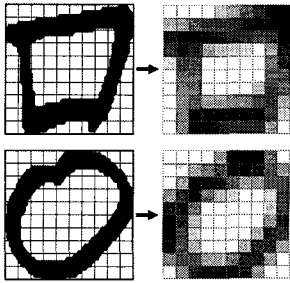


그림 1. 메쉬 특징 비교

나. 투영(projection) 특징

수직 투영(vertical projection)과 수평 투영(horizontal projection)으로 구분할 수 있으며, 영상의 수직 및 수평 방향으로 누적된 픽셀 개수를 특징으로 이용한다. 영상을 $M \times N$ 으로 정규화하였을 경우 총 $(M+N)$ 개의 특징이 추출되며, 이런 투영 특징은 특징의 개수가 작으며 약간의 위치 이동이나 획의 기울어짐에는 강인한 특성을 보이는 반면 전체적인 획전에는 둔감한 특징을 가지고 있다.

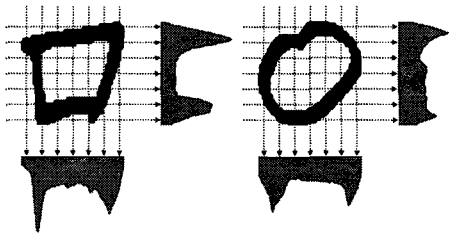


그림 2. 투영 특징 비교

다. 교차 거리(cross distance) 특징

영상의 상, 하, 좌, 우, 4방향의 외곽으로부터 획을 만날 때까지의 거리를 특징으로 이용한다. 영상을 $M \times N$ 으로 정규화하였을 경우 총 $(2 \times M + 2 \times N)$ 개의

특징이 추출되며, 이런 교차 거리 특징은 특징의 개수가 작으며 문자의 방향별 문자의 외곽 정보를 자세히 알 수 있으나 내부 구조에 대한 정보는 검출할 수 없는 단점이 있다.

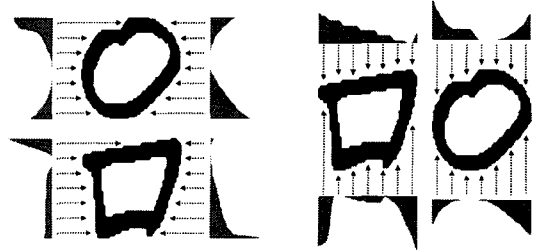


그림 3. 교차 거리 특징

2.3 지역, 위상적 특징 추출 방법

지역, 위상적 특징 추출 방법은 문자의 기하 및 위상적인 특징을 추출하는 방법이다. 이러한 특징에는 방향 코드(directional codes), 다각형 근사법, 컨볼루션 특징, 획 분할 등이 있다[6,11]. 본 논문에서 제안하는 이웃 각도 히스토그램 특징은 지역, 위상적인 특징을 이용하는 방법으로, 'ㄱ'과 'ㅇ'을 인식하는데 적용되었다.

3. 이웃 각도 히스토그램 특징

3.1 이웃 각도 히스토그램

그림 4는 완전한 사각형과 원형의 이웃 각도 분포를 비교하고 있다. 여기서, 이웃 각도(neighbor angle)는 획을 이루는 한 픽셀(중심 픽셀)과 그 픽셀과 같은 거리에 있는 두 개의 이웃 픽셀 사이의 각도로 정의한다.

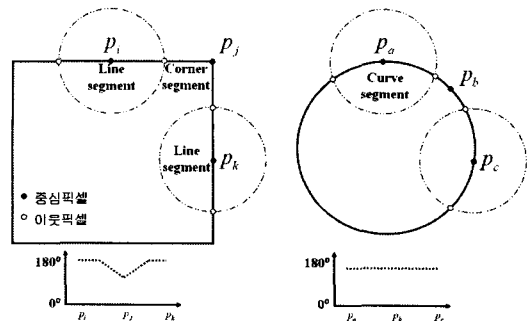


그림 4. 이웃 각도 분포 비교

그림 4에서 알 수 있는 바와 같이 ‘口’과 ‘o’ 이미지의 형태를 분석해 보면, 개개의 필기 습관에 따라 약간씩 다르지만 일반적으로 ‘口’의 경우 4방향의 모서리 부분(corner segment)이 있으며 그 외 부분에 대해서는 직선 부분(line segment)으로 획을 형성하고 있다. 이와 대조적으로 ‘o’은 전반적으로 원형에 가까운 곡선 부분(curve segment)으로 획의 모양을 형성하고 있다.

즉, ‘口’과 같은 사각형 구조에서는 모서리를 중심으로 이웃 각도 분포의 변화가 있는 반면, ‘o’과 같은 원형 구조에서는 일률적인 각도 분포를 나타냄을 알 수 있다. 이러한 획의 모양에 근거하여 ‘口’과 ‘o’을 효과적으로 구분하기 위한 특징으로 이웃 각도 히스토그램 $H_r(\theta)$ 을 다음과 같이 정의한다.

$$H_r(\theta) = WELL \left\{ \begin{array}{l} P_j | \theta - \epsilon < \angle(P_{j1}, P_j, P_{j2}) \leq \theta, \\ r = \|P_{j1} - P_j\| = \|P_j - P_{j2}\|, \\ P_j, P_{j1}, P_{j2} \in I \end{array} \right\} \quad (1)$$

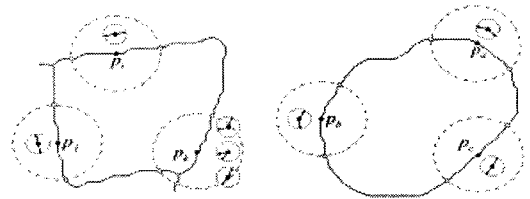
$$\theta \in \{\theta_i | 0 \leq i \leq c, c \in Z\} \quad (2)$$

$$\epsilon = \frac{180^\circ}{c} \quad (3)$$

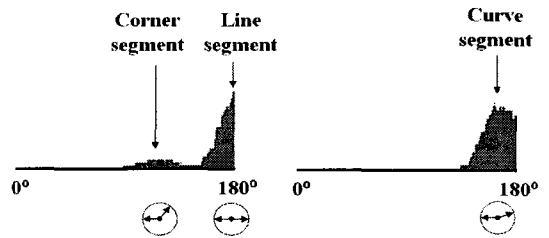
여기서, $WELL(\cdot)$ 은 집합의 원소들의 개수, $\angle(P_{j1}, P_j, P_{j2})$ 는 $\overrightarrow{P_j P_{j1}}$ 과 $\overrightarrow{P_j P_{j2}}$ 사이의 각도인 이웃 각도, $\|\cdot\|$ 는 두 점 사이의 유클리디안 거리(euclidean distance), I 는 획을 구성하는 픽셀들의 집합, N 은 이미지의 크기, c 는 각도 분할 개수, ϵ 은 분할된 각의 범위로 정밀도를 나타낸다. 예를 들어, 1°단위로 이웃 각도 히스토그램을 구하는 경우 분할된 각의 범위 $\epsilon=1^\circ$, 각도 분할 개수 $c=180$ 이 된다. 즉, $H_r(\theta)$ 는 중심 픽셀 P_j 와 중심 픽셀에서 거리 r 만큼 떨어진 획 위의 두 픽셀 P_{j1}, P_{j2} 사이의 각도가 $\theta - \epsilon$ 과 θ 사이인 중심 픽셀들의 개수를 의미한다.

그림 5는 획을 구성하는 픽셀과 그 이웃 픽셀과 이루는 각도를 나타내고 있으며, 그림 6은 획을 구성하는 모든 픽셀에 대한 이웃 각도 히스토그램의 분포를 나타내고 있다.

그림 5에서 살펴볼 수 있는 바와 같이 ‘口’의 경우 P_i, P_j 와 같이 직선 영역에 해당하는 픽셀이 다수를 이루며, P_k 와 같이 모서리 영역에 해당하는 픽셀 분포도 어느 정도 존재함을 알 수 있다. P_k 와 같은 픽셀은 세션화 과정에서 발생된 잔가지에 의한 영향으로



(a) ‘口’ 획의 이웃 각도 (b) ‘o’ 획의 이웃 각도
그림 5. 획을 구성하는 픽셀들의 이웃 각도 비교



(a) ‘口’ 이웃 각도 히스토그램 (b) ‘o’ 이웃 각도 히스토그램

그림 6. 이웃 각도 히스토그램 분포 비교

3가지 경우의 각도가 산출됨을 알 수 있다. 또한, ‘o’의 경우 P_c 와 같이 직선 영역에 해당되는 픽셀도 어느 정도 존재하며, P_a, P_b 와 같이 곡선 영역에 해당하는 픽셀이 많은 분포를 이루고 있음을 알 수 있다.

3.2 최적의 이웃 픽셀 반경 설정

중심 픽셀을 기준으로 일정 거리에 존재하는 이웃 픽셀을 선정하기 위해서는 중심 반경 r 이 설정되어야 한다. 이 중심 반경에 따라 ‘口’과 ‘o’ 이미지에서 추출되는 각도 분포도 변화하므로 두 패턴을 구분할 수 있는 특징도 달라진다.

그림 7은 가로와 세로의 길이가 같은 최소 외접 사각형(MBR : Minimum boundary rectangle)으로 설정된 ‘o’ 이미지의 가로 또는 세로 길이의 20%(0.2)와 30%(0.3) 크기로 중심 반경을 설정하였을 경우에 추출되는 이웃 각도가 각각 157°, 145°가 되는 것을 보여준다.

이 중심 반경이 너무 작거나 너무 크면 ‘口’과 ‘o’ 이미지에서 추출되는 각도 분포가 유사하게 되어 서로 구분되는 특징이 없어지게 된다. 그러므로 효과적으로 이웃 각도 특징을 추출하기 위한 적절한 중심 반경의 설정은 매우 중요하다. 본 논문에서는 이 중심 반경을 설정하기 위한 방법으로 클래스 내 거리

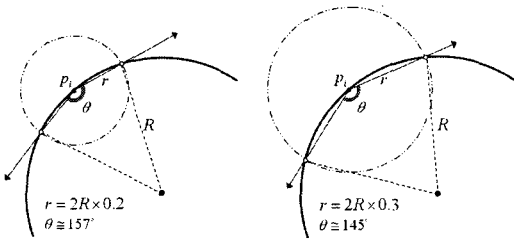


그림 7. 중심 반경 변경에 따른 이웃 각도 변화 비교

(Inner class distance)는 최소화하고 클래스 간 거리 (Between class distance)는 최대화하는 방법으로 중심 반경을 설정하였다. 즉, 특정 중심 반경 값에 의해서 추출되는 이웃 각도 히스토그램은 같은 패턴인 경우(‘ㄱ’ 또는 ‘ㅇ’)에는 서로 유사한 분포를 나타내고 다른 패턴(‘ㄱ’과 ‘ㅇ’)인 경우에는 서로 상이한 분포를 나타내도록 최적 반경 설정 기준 $R(M, Y)$ 을 다음과 같이 정의한다.

$$R(M, Y) = \frac{D_h(M, Y)}{D_{\max}(M)} + \frac{D_h(M, Y)}{D_{\max}(Y)} \quad (4)$$

$$D_{\max}(M) = \max_{i, j \in M} PVER_i - jPVER \quad (5)$$

$$D_h(M, Y) = \max_{m \in M} \min_{y \in Y} PVER_m - yPVER \quad (6)$$

$$D_h(M, Y) = \min_{m \in M} \max_{y \in Y} PVER_m - yPVER \quad (7)$$

여기서, m, y 는 각각 집합 M, Y 의 원소로 이웃 각도 히스토그램을 의미하여, $PVER_m - yPVER$ 는 두 집합내 원소 m 과 y 사이의 유클리디안 거리(euclidean distance), $D_h(M, Y)$ 는 하우스도르프 거리, $D_h(M, Y)$ 는 변형된 하우스도르프 거리, $D_{\max}(M)$ 는 집합 내 원소간의 최대 거리를 나타낸다.

하우스도르프 거리는 위상 수학의 한 개념으로 유한 개의 점을 포함하는 두 개의 집합 M 과 Y 의 원소들의 유사 정도를 표시하기 위한 Min-Max 거리로써 많은 연구에서 형태 정합(Shape matching)에 사용되어 왔다[12,13]. 본 논문에서는 두 개의 집합 M 과 Y 원소들의 비 유사도를 측정하기 위한 Max-Min 거리로 변형된 하우스도르프 거리 $D_h(M, Y)$ 를 정의하였다. 즉, ‘ㄱ’ 이미지 집합 M 과 ‘ㅇ’ 이미지 집합 Y 에서 $R(M, Y)$ 가 최대값을 가지는 중심 반경 r 을 두 클래스를 가장 잘 분리할 수 있는 최적 중심 반경으로 설정한다.

4. 실험 및 결과 분석

4.1 Data Set

실험에 사용된 한글 자소 ‘ㄱ’과 ‘ㅇ’ 이미지는 총 3개의 SET에 368개의 이미지로 SET1은 pc92DB의 문자에서 추출한 100개의 자소 이미지, SET2는 11명의 연구실 사람들이 싸인펜으로 10자씩 A4지에 작성한 220개 자소 이미지, SET3은 바탕, 굴림, 돋움, 궁서 등 12종의 글꼴을 기울임 속성을 포함하여 10point로 작성한 48개 자소 이미지로 구성되어 있다.

SET2와 SET3의 자소 이미지는 HP ScanJet 6350C 스캐너를 사용하였으며 300DPI 해상도로 스캔하여 획득한 후 수동으로 분할하여 만들었으며 획득된 자소 이미지에서 가로 및 세로의 길이는 대략 30~80 pixel 정도의 크기를 가진다. 총 368개의 문자 중 임의로 1/3(122개)를 택하여 학습에 사용하였고 나머지 2/3(246개)는 학습된 인식기에 대한 성능 평가에 사용하였다.

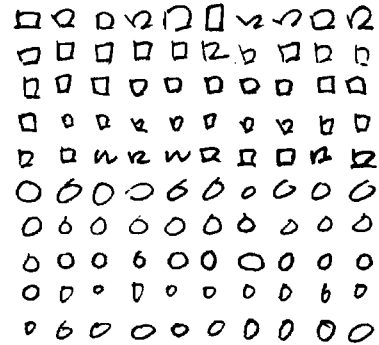


그림 8. 실험 데이터 예

4.2 실험 방법

본 논문에서 제시하는 이웃 각도 히스토그램 특징의 인식 성능을 검증하기 위한 실험 방법으로 K-means Clustering에 의한 인식 방법[14]을 사용하였다. 최적화된 인식 성능을 획득하기 위하여 SVM (Support Vector Machine), 인공 신경망(Neural Network) 등 K-means Clustering 이외에 여러 가지 방법들이 소개되고 있으나, 본 논문에서는 인식기 자체의 성능보다는 추출되는 특징의 공정한 비교를 위하여 비교되는 특징간의 참조 패턴 개수를 동일하게 설정할 수 있도록 K-means Clustering에 의한 인식

방법을 선택하였다.

또한, 비교 특징은 메쉬 특징과 수직수평투영 및 교차거리 특징을 사용하였으며, 특징들 간의 비교 방법으로 특징 개수에 의한 비교, 참조 패턴 개수에 의한 비교, '口' 과 'o'에 대한 비교를 수행하여 인식 성능을 평가하였다. 마지막으로 최적화된 이웃 각도 특징을 추출하기 위하여 최적 반경 설정 기준에 따른 분석을 수행하였으며, 실험으로 최적 반경 설정 기준의 타당성을 증명하였다.

4.3 K-means Clustering

K-means 알고리즘에 의한 클러스터링은 임의의 c개의 패턴으로 초기 클러스터 중심 패턴을 설정한 후 입력패턴과 클러스터의 중심패턴과의 거리가 최소가 되는 클러스터에 입력패턴을 멤버로 포함시키면서 k개의 멤버들의 평균으로 클러스터의 중심값을 조정하면서 입력패턴을 분류하는 방법으로 그 알고리즘은 표 1과 같다.

표 1. K-means Clustering 알고리즘

[단계1]	입력패턴 집합 $X=\{x_1, x_2, \dots, x_N\}$, 참조패턴 집합 $V=\{v_1, v_2, \dots, v_c\}$, N : 입력패턴 수, c : 클러스터 수, T : 총 반복회수, θ : 허용오차값 설정
[단계2]	$V_0=\{v_{1,0}, v_{2,0}, \dots, v_{c,0}\}$: 0과 1사이의 난수로 초기 참조패턴 벡터 설정
[단계3]	for $t=1,2,3,\dots,T$ for $k=1,2,3,\dots,N$ a. 승자 참조패턴 선택 $x_k - v_{i,t} = \min\{x_k - v_{j,t}\}, (1 \leq j \leq c)$ b. 승자 참조패턴 벡터 수정 $v_{i,t+1} = \frac{1}{N_j} \sum_{x_j \in S_{k0}} x_j$ 여기서, $S_j(t)$ 는 시간 t에서 참조패턴 $v_{i,t}$ 에 속하는 패턴들의 집합, N_j 는 $S_j(t)$ 의 크기 c. next k
[단계4]	반복에 따른 오차 계산 $E_t = V_t - V_{t-1} = \sum v_{kt} - v_{kt-1}$
[단계5]	만약, $E_t \leq \theta$ 이면 학습 중단 아니면 학습률 조정, next t

4.4 인식 성능 비교 분석

가. 특징 개수별 비교

그림 9~11은 메쉬 특징, 투영 및 교차 거리 특징, 이웃 각도 특징의 개수별 인식을 비교이다. 메쉬 특징은 원영상을 각각 $10 \times 10(100)$, $13 \times 14(182)$, $15 \times 15(225)$, $20 \times 20(400)$ 의 메쉬 블록으로 나누어 특징을 추출하였다. 10×10 보다 더 작은 블록으로 나눌 경우에는 이미지의 변별력이 너무 떨어지므로 최소 크기를 10×10 으로 설정하였다. 투영 및 교차 거리 특징은 총 6개의 특징(수평투영+수직투영+좌방향 교차거리+우방향 교차거리+상방향 교차거리+하방향 교차거리)을 결합하였으며, 각각 $10 \times 10(60)$, $15 \times 15(90)$, $20 \times 20(120)$, $30 \times 30(180)$ 크기의 투영 및 교차 거리 특징을 추출하였다. 각도 특징은 원영상의 정규화 크기와는 무관하며 추출 각도의 분할 범위를 각각 $10^\circ(18)$, $6^\circ(30)$, $3^\circ(60)$, $2^\circ(90)$, $1^\circ(180)$ 로 하여 특징을 추출하였다.

그림 9~11에서 알 수 있는 바와 같이 메쉬 특징은 어느 정도의 인식 성능은 나타내지만 추출되는 특징의 개수가 매우 많은 단점이 있으며, 투영 및 교차거리 특징은 특징의 개수는 비교적 적은 편이지만 이에 상응하여 인식 성능이 떨어지는 단점이 있는 것을 알 수 있다. 이것은 일반적으로 관찰되는 특징의 개수 및 인식 성능의 비례 관계를 나타내고 있다. 하지만 제안하는 방법은 추출되는 특징의 개수도 다른 특징에 비하여 월등히 작은 반면 가장 좋은 이상적인 인식 성능을 나타내었다. 또한 특징의 개수에 상관없이 비교적 일관된 성능을 나타내었고 학습된 데이터에 대한 인식률과 학습되지 않은 데이터에 대한 인식

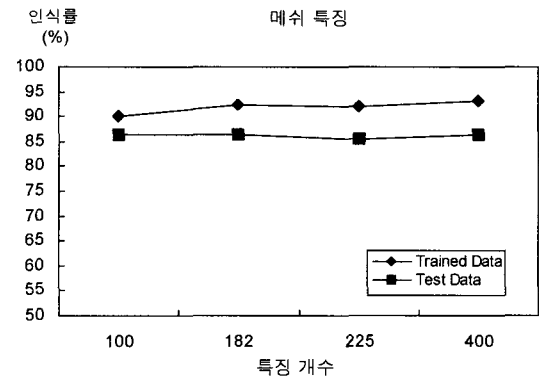


그림 9. 메쉬 특징

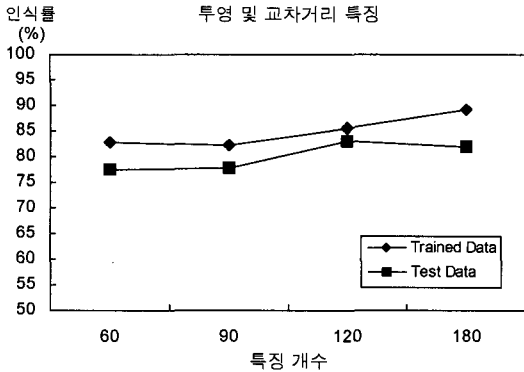


그림 10. 투영+교차거리 특징

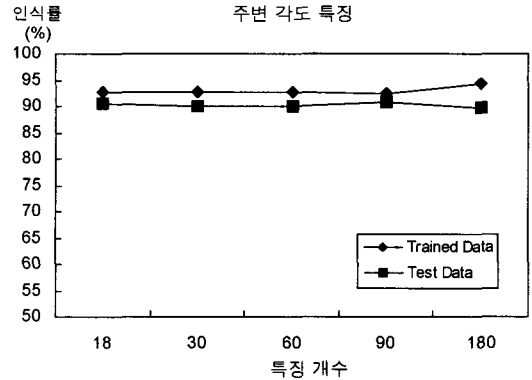


그림 12. 이웃 각도 특징

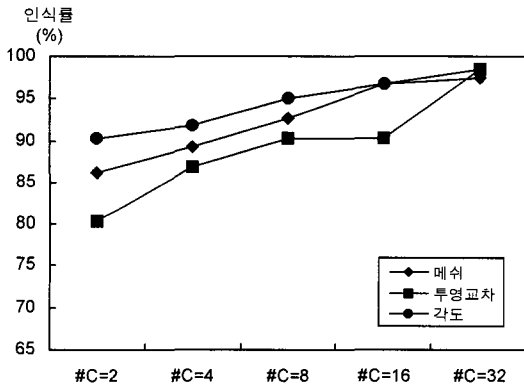


그림 11. 참조 패턴 개수에 따른 특징별 학습 데이터 인식률 비교

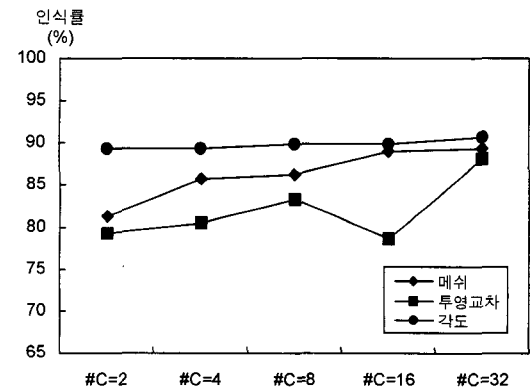


그림 13. 참조 패턴 개수에 따른 특징별 테스트 데이터 인식률 비교

률의 차이도 다른 특징에 비해 작다는 것을 알 수 있으며 이것은 미지의 패턴에 대한 일반성 및 안정성이 우수하다는 것을 알 수 있다.

나. 참조 패턴 개수별 비교

그림 12와 그림 13은 참조 패턴의 개수에 따른 특징별 학습 데이터 및 테스트 데이터의 인식률을 비교하고 있다. 즉, K-means Clustering으로 인식기를 생성할 때 참조 패턴을 각각 2개, 4개, 8개, 16개, 32개로 변화시키면서 클러스터링을 수행하였으며, 이것은 'ㄱ'과 'ㅇ'에 대한 참조 패턴이 각각 1개, 2개, 4개, 8개, 16개가 존재함을 의미한다. 그림에서 알 수 있는 바와 같이, 일반적으로 학습된 데이터에 대한 인식률이 학습되지 않은 데이터에 대한 인식률보다 높게 나타나게 되며, 학습된 데이터의 경우 참조 패턴의 개수가 많을수록 인식률도 어느 정도 향상되는 것을 관찰할 수 있다. 특히 제안하는 방법은 테스트 데이

터의 인식 성능에서 살펴볼 수 있는 바와 같이 참조 패턴의 개수에 민감하게 영향을 받지 않으며 안정된 인식 성능을 나타냄을 알 수 있다.

다. 'ㄱ', 'ㅇ' 인식 성능 비교

그림 14는 'ㄱ', 'ㅇ' 각각에 대한 인식 성능을 비교하였다. 전반적으로 살펴볼 때 변형 패턴이 많은 'ㄱ' 보다는 'ㅇ'이 더욱 좋은 인식 성능을 나타냄을 알 수 있으며, 학습된 데이터의 경우 메쉬 특징에 의한 방법이 'ㅇ'의 인식에 가장 좋은 성능을 나타내었지만, 테스트 데이터에 대해서는 제안하는 방법이 가장 좋은 성능을 나타냄을 알 수 있다.

라. 최적 중심 반경 설정

그림 15는 'ㄱ'과 'ㅇ' 클래스를 각각 M, Y 라고 할 때 M, Y 각각 10개의 대표적인 패턴을 선정하여 자소 영상에서 최소 외접 사각형의 가로 또는 세로

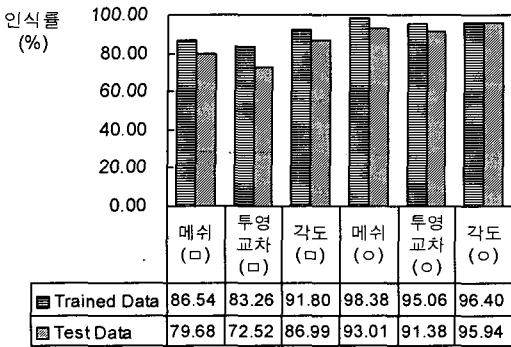


그림 14. 'ㄱ', 'ㅇ' 분할 인식 성능 비교

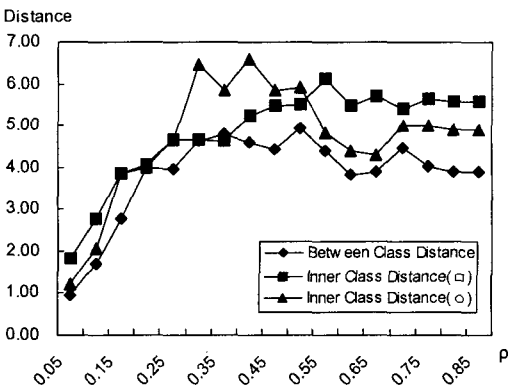


그림 15. 'ㄱ'과 'ㅇ' 클래스간 거리와 클래스내 거리 비교

길이의 5%~85% 크기(p)로 중심 반경이 변화할 경우 클래스 간 거리 $D_k(M, Y)$ 와 각각의 내부 거리 $D_{max}(M)$, $D_{max}(Y)$ 와의 관계를 보여주고 있다.

그림에서 알 수 있는 바와 같이 중심 반경을 전체 영상의 5%(0.05) 정도 크기로 작게 설정하면 클래스 내 거리도 작아지지만 클래스 간 거리도 작아지는 것을 알 수 있다. 즉, 중심 반경을 작게 하여 추출된 이웃 각도 특징은 같은 클래스의 패턴에 대한 특징도 서로 유사하게 나타날 뿐만 아니라 다른 클래스 간('ㄱ'과 'ㅇ') 특징도 서로 유사함을 의미하며 이때의 중심 반경을 기준으로 추출된 각도 특징으로서 두 클래스를 구별할 수 있는 차이점이 별로 없으므로 바람직한 선택이 되지 못한다. 또한, 중심 반경을 전체 영상의 30%~40%(0.30~0.40) 크기로 하였을 경우, 'ㅇ'의 클래스 내 거리가 매우 크게 나타남을 알 수 있다. 이것은 이 정도의 중심 반경으로 이웃 각도 특징을 추출할 경우 같은 'ㅇ' 패턴이지만 매우 상이한 특징이 추출됨을 알 수 있으며 이것 또한 바람직

하지 못한 결과가 된다. 따라서 두 개의 클래스를 가장 잘 구분할 수 있는 적절한 각도 벡터를 추출하기 위해서는 적절한 중심 반경을 설정하는 것이 매우 중요하다.

그림 16은 그림 15를 바탕으로 클래스 내 거리는 최소화하고 클래스 간 거리는 최대화하는 방법으로 최적 반경 설정 기준 $R(M, Y)$ 을 산출한 결과이며, 그림 17, 18에서는 실제 중심 반경을 10%, 20%, 30%로 다르게 설정하여 추출된 이웃 각도 특징을 바탕으로 인식 성능을 비교하고 있다.

결과에서 알 수 있듯이 이웃 각도 특징을 추출하기 위해서 중심 반경을 20% 정도 크기로 설정하였을 경우 가장 최적화된 이웃 각도 특징을 추출할 수 있는 중심 반경이 됨을 알 수 있다. 그리고 변형된 하우스도르프 거리에 의해 클래스 내 거리는 최소화하고 클래스 간 거리는 최대화하는 최적 반경 설정 기준 $R(M, Y)$ 이 이웃 각도 특징을 추출하기 위한 중심 반경을 설정하는 객관적 지표가 될 수 있음을 알 수 있다.

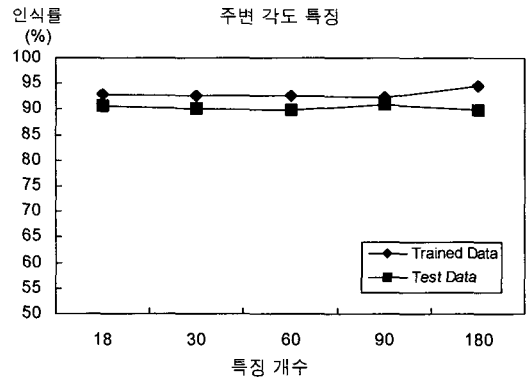


그림 16. 이웃 각도 특징

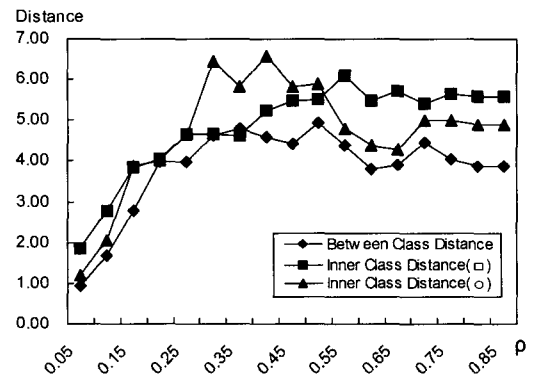


그림 17. 'ㄱ'과 'ㅇ' 클래스간 거리와 클래스내 거리 비교

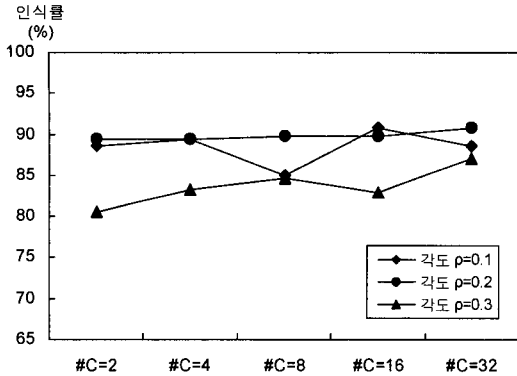


그림 18. 중심 반경에 따른 각도 특징의 인식 성능 비교(테스트 데이터)

마. 오인식 패턴 분석

그림 19와 그림 20은 각각 'ㄱ'과 'ㅇ'의 대표적인 오인식 패턴에 대한 원영상과 정규화된 세선화 영상의 예이다.

그림에서 살펴볼 수 있는 바와 같이 'ㄱ' 오인식 패턴의 경우 모서리각이 불분명하여 'ㅇ'과 유사한 형태의 세선화 결과를 나타내는 것을 알 수 있으며, 'ㅇ' 오인식 패턴의 경우 원영상이 너무 작아서 획이

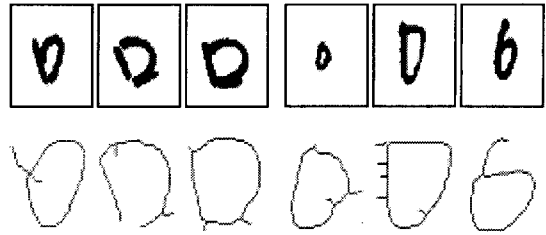


그림 19. 'ㄱ' 오인식 패턴 원영상 및 세선화 그림 20. 'ㅇ' 오인식 패턴 원영상 및 세선화

상대적으로 두꺼워지거나 획 근처에 존재하는 잡영에 의해서 세선화가 올바르게 수행되지 못하여 잔가지 등이 발생한 경우 이웃 각도 벡터가 'ㄱ'과 유사한 형태로 산출됨으로써 오인식이 발생한 경우로 분석된다.

표 2~4에서는 제안하는 알고리즘의 인식 성능을 비교 분석하여 요약하였다.

5. 결론 및 향후 연구 방향

본 논문에서는 한글 문자의 자소 중 구분하기가 모호하여 인식률이 낮은 'ㄱ'과 'ㅇ'을 효율적인 인식

표 2. 특징별 인식 성능 비교표

특징	특징 개수	학습 데이터					평균	테스트 데이터					평균
		참조패턴 개수 (#C)						참조패턴 개수 (#C)					
		2	4	8	16	32		2	4	8	16	32	
매쉬 특징	100	83.6	86.9	87.7	93.4	98.4	90.0	83.7	83.7	85.8	88.6	90.7	86.5
	182	86.1	89.3	92.6	96.7	97.5	92.4	81.3	85.8	86.2	89.0	89.4	86.3
	225	86.9	86.9	93.4	95.1	98.4	92.1	81.3	82.1	86.6	87.0	90.2	85.4
	400	86.1	90.2	94.3	97.5	98.4	93.3	85.0	83.3	84.6	87.8	91.5	86.4
	평균	85.7	88.3	92.0	95.7	98.2	92.0	82.8	83.7	85.8	88.1	90.5	86.2
투영+교차거리 특징	60	73.0	76.2	77.9	92.6	94.3	82.8	78.9	78.9	75.2	75.2	79.7	77.6
	90	70.5	76.2	84.4	86.9	93.4	82.3	79.7	70.3	76.0	76.4	86.6	77.8
	120	77.0	73.8	89.3	91.0	96.7	85.6	82.5	78.5	84.6	84.6	85.0	83.0
	180	80.3	86.9	90.2	90.2	98.4	89.2	79.3	80.5	83.3	78.5	88.2	82.0
	평균	75.2	78.3	85.5	90.2	95.7	85.0	80.1	77.1	79.8	78.7	84.9	80.1
각도 특징	18	88.5	89.3	91.8	96.7	97.5	92.8	91.5	91.9	90.7	90.2	88.6	90.6
	30	87.7	87.7	94.3	95.9	97.5	92.6	91.1	89.4	87.8	89.8	91.9	90.0
	60	86.9	90.2	93.4	94.3	98.4	92.6	91.5	88.6	89.8	89.8	89.8	89.9
	90	91.0	90.2	91.0	93.4	95.9	92.3	90.2	91.5	90.7	88.2	93.9	90.9
	180	90.2	91.8	95.1	96.7	98.4	94.4	89.4	89.4	89.8	89.8	90.7	89.8
	평균	88.9	89.8	93.1	95.4	97.5	93.0	90.7	90.2	89.8	89.6	91.0	90.2

표 3. 'ㄹ', 'ㅇ' 분할 인식 성능 비교

구분	특징별	학습 데이터						테스트 데이터					
		참조패턴 개수 (#C)					평균	참조패턴 개수 (#C)					평균
		2	4	8	16	32		2	4	8	16	32	
미음 (ㄹ)	메쉬	77.0	80.3	86.9	93.4	95.1	86.5	74.8	75.6	80.5	82.1	85.4	79.7
	투영+교차거리	72.1	80.3	83.6	83.6	96.7	83.3	72.4	69.9	74.0	65.0	81.3	72.5
	각도	88.5	86.9	93.4	95.1	95.1	91.8	87.8	83.7	91.1	89.4	82.9	87.0
	미음평균	79.2	82.5	88.0	90.7	95.6	87.2	78.3	76.4	81.8	78.9	83.2	79.7
이용 (ㅇ)	메쉬	95.1	98.4	98.4	100	100	98.4	87.8	95.9	91.9	95.9	93.5	93.0
	투영+교차거리	88.5	93.4	96.7	96.7	100	95.1	86.2	91.1	92.7	91.9	95.1	91.4
	각도	93.4	96.7	95.1	98.4	98.4	96.4	96.8	96.8	94.3	95.1	96.8	95.9
	이용평균	92.3	96.2	96.7	98.4	99.5	96.6	90.2	94.6	93.0	94.3	95.1	93.4

표 4. 중심 반경에 따른 각도 특징의 인식 성능 비교표

	학습 데이터						테스트 데이터					
	참조패턴 개수 (#C)					평균	참조패턴 개수 (#C)					평균
	2	4	8	16	32		2	4	8	16	32	
각도 p=0.1	89.3	94.2	95.1	93.4	96.7	93.7	88.6	89.4	85.0	90.7	88.6	88.5
각도 p=0.2	90.2	91.8	95.1	96.7	98.4	94.4	89.4	89.4	89.8	89.8	90.7	89.8
각도 p=0.3	84.4	86.9	91.0	91.8	95.1	89.8	80.5	83.3	84.6	82.9	87.0	83.7
평균	88.0	91.0	93.7	94.0	96.7	92.7	86.2	87.4	86.5	87.8	88.8	87.3

하기 위한 특징 추출 방법을 제시하였다. 먼저 원 영상이 되는 자소 이미지를 세선화하여 획 성분을 추출하고 획을 구성하는 각 픽셀과 이웃하는 두 픽셀 사이의 각도 성분으로 구성된 이웃 각도 히스토그램을 두 개의 자소를 구분할 수 있는 특징으로 사용하였다.

이때, 이웃 각도를 추출하기 위한 중심 반경의 설정을 위하여 변형된 하우스도르프 거리를 사용한 최적 중심 반경 설정 기준을 정의하였으며, 중심 반경을 다양하게 변화시켜가면서 최적 중심 반경 설정 기준을 조사한 결과 원 영상 크기의 20%(0.2) 정도로 중심 반경을 설정하는 것이 최적의 이웃 각도 특징을 추출하는 방법임을 알 수 있었다.

제안하는 이웃 각도 특징 추출 방법을 사용하여 'ㄹ', 'ㅇ'을 인식한 결과 다른 특징 추출 방법에 비하여 훨씬 작은 특징으로도 가장 높은 인식 성능(90%)을 나타내었으며, 특징의 개수 및 참조 패턴의 개수에 민감하게 영향을 받지 않는 안정적인 성능을 나타내었다. 그리고 학습되지 않은 패턴에 대해서도 우수한 인식 성능을 보였으며 이것은 미지의 패턴에 대한 일반성을 보장하는 효과적인 특징 추출 방법임을 알 수 있었다.

향후 연구 방향으로는 잡영에 의한 영향을 최소화할 수 있는 세선화 방법 등 획 영상의 개선을 위한 전처리 방법에 대한 연구와 한글 인식에 있어서 'ㄹ'

과 'ㅇ'을 구분하는 후처리 방법으로서의 실제적인 적용 방법에 대한 연구, 원형 고리 특징 등 다른 특징 추출 방법과 결합하여 보다 나은 특징 벡터를 생성할 수 있는 방법에 대한 연구를 수행할 계획이다. 또한 입력 패턴의 변형에 적응적인 참조 패턴을 생성할 수 있도록 인공 신경망을 적용한 인식기를 사용하여 다양한 변이 패턴을 가지고 있는 'ㄹ'에 대한 인식 성능을 높일 수 있는 방법에 대한 연구도 추진되어야 할 것이다.

참 고 문 헌

- [1] 박창순, 김두영, "오프라인 필기체 숫자 인식을 위한 다양한 특징들의 비교 및 인식을 개선 방안," 한국 정보처리학회 논문지, 제3권 4호, 1996년 7월.
- [2] 이성환, 박희선, "고리 투영을 이용한 위치, 크기 및 회전 변형에 무관한 한글 문자 인식," 인지과학 Korean Journal of Cognitive Science, VOL. 3, No.1, pp.139-160, 1991년.
- [3] 조성배, 김진형, "온라인 필기문자의 인식을 위한 은닉 마르코프 모델과 신경망 분류기의 결합," 한국 정보과학회 논문지, VOL.20, No.3, 1993년 3월.

- [4] B. Y. Choi, S. Y. Bang, "Constraints on hand-written Korean characters to improve the machine readability," *International Journal of Industrial Ergonomics* 26 (2000) 549-559.
- [5] N. Arica, F. T. Yarman-Vural, "An overview of character recognition focused on off-line handwriting," *IEEE Trans, Systems, Man and Cybernetics, Part C, Volume: 31, Issue: 2, pp.216-233, May 2001.*
- [6] 고태석, 김종렬, 정규식, "오프라인 필기체 한글 자소 인식에 있어서 특징성능의 비교," *한국인지과학회논문집, 제7권 제1호, pp.57-74, 1996년.*
- [7] 김호연, 서장원, 권재욱, 김진형, "필기 한글 인식에 관한 연구 : KAIST의 연구를 중심으로," *한국인지과학회논문집, 제6권 제4호, pp.5-20, 1995년.*
- [8] Pyeoung Kee Kim, Jong Kook Lee, Hang Joon Kim, "Handwritten Korean Character Recognition by Stroke Extraction and Representation," *IEEE TENCON, 1993년.*
- [9] 장석진, 강선미, 김혁구, 노우식, 김덕진, "자소 인식 신경망을 이용한 한글 문자 인식에 관한 연구," *전자공학회논문지, 제31권, pp.81-98, 1994.*
- [10] MILAN SONICA, VACLAY HLAVAC, ROGER BOYLE, *Image Processing, Analysis, and Machine Vision 2nd Edition*, PWS Publishing, pp.129-130.
- [11] Seon-Hwa Jeong, Yun-Seok Nam, Hye-Kyu Kim, "Non-similar candidate removal method for off-line handwritten Korean character recognition," *IEEE ICDAL 2003년.*
- [12] D.Huttenlocher, D.Klanderman, A.Rucklige, "Comparing Images using The Hausdorff Distance," *IEEE Tans. on PAMI, Vol.15, No.9, pp.850-863, Sep., 1993.*
- [13] 전춘, 이주신, "이동물체 추적을 위한 실시간 하우스도르프 정합 알고리즘," *정보처리학회 논문지B, 제9-B권, 제6호, pp.707-714, 2002년 12월.*
- [14] Arun D.Kulkarni, *Computer Vision and Fuzzy-neural systems, pp.357-358, Prentics Hall, 2001.*



김도현

2001년 부산대학교 전자계산학과 졸업
 2003년 부산대학교 대학원 전자계산학과 석사 졸업
 2004년 현재 부산대학교 대학원 컴퓨터공학과 박사 재학중

관심분야 : 패턴인식, 영상처리 및 컴퓨터비전, 신경회로망 및 퍼지 시스템 등



장원두

2003년 부산대학교 정보컴퓨터공학부 졸업
 2004년 현재 부산대학교 대학원 컴퓨터공학과 재학중

관심분야 : 패턴 인식, 인공신경회로망, 컴퓨터 비전, 생태 환경 정보 시스템



김하영

2003년 부산대학교 정보컴퓨터공학부 졸업
 2004년 현재 부산대학교 대학원 컴퓨터공학과 재학중

관심분야 : 영상처리 및 컴퓨터 비전, 문자인식, 웨이블릿, 패턴 인식 등



차의영

1979년 경북대학교 전자공학과 졸업
 1982년 서울대학교 대학원 계산통계학과 석사졸업
 1998년 서울대학교 대학원 컴퓨터공학과 박사졸업

1995년~1996년 University of London 방문교수
 1985년~현재 부산대학교 컴퓨터공학과 교수
 관심분야 : 컴퓨터비전, 신경회로망, 웨이블릿, 워터마크