

획의 방향 코드 조합에 의한 오프라인 필기체 숫자 인식

이찬희⁰ 이상훈⁰⁰ 장수미⁰⁰ 정순호

부경대학교 전자계산학과⁰ 부경대학교 전산정보학과⁰⁰ 부경대학교 전자컴퓨터정보통신 공학부
{chlee⁰, shlee⁰⁰, smjang⁰⁰}@aisol.pknu.ac.kr shjung@pine.pknu.ac.kr

Off-line Handwritten Digit Recognition Using Combination of stroke direction codes

Chan-Hee Lee⁰ Sang-Hoon Lee⁰⁰ Soo-Mi Jang⁰⁰ Soon-Ho Jung

Dept. of Computer Science, Pukyong National University⁰

Dept. of Computer and Information, Pukyong National University⁰⁰

Faculty of Electronics, Computer, Telecommunication Engineering, Pukyong National University

요 약

본 논문은 오프라인 필기체 숫자 인식을 위하여 SOG* 세선화와 방향 코드 생성만으로 전처리를 단순화 하여 효율을 높이는 새로운 방법을 제안한다. 본 실험의 객관적 검증을 위해 Concordia 대학교 등의 여러 기관의 필기체 숫자 데이터베이스에 대하여 실험한 결과 98.85% 이상의 인식률을 나타내어 단순한 전처리로 높은 인식률을 얻음으로써 효율성이 높음을 알 수 있다.

1. 서론

오프라인 필기체 문자 인식의 기본이 되는 오프라인 필기체 숫자 인식에 있어 인식 성능 향상을 위하여 전처리 단계에서 패턴의 형태를 다양하게 변형하는 방법이 사용된다. 이 변형 방법은 크게 선형 형태 변형(Linear Shape Variation)과 비선형 형태 변형(Non-linear Shape Variation)으로 분류할 수 있으며 선형 형태 변형으로는 이동, 신축, 회전, 밀림 등이 포함되며, 비선형 형태 변형 방법으로는 점밀도, 선밀도, 획밀도 등이 포함된다[1,2].

이에 반해, 신경 회로망을 이용한 방법은 신경 회로망이 갖는 적응적 특성으로 인해 많은 변형을 필요로 하지 않는다. 그러나 기존 신경 회로망을 이용한 방법은 특징 추출의 미비로 다양한 형태를 흡수하지 못하는 문제점이 발생하거나, 인식률을 높이기 위해 전처리 과정에서 많은 변형을 사용함으로써 전처리 과정이 복잡해지고 인식기의 입력도 많아지는 단점을 가지고 있다.

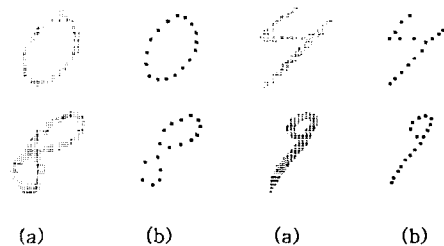
따라서 본 논문에서는 필기체 숫자에서 방향 코드라는 단순한 특징을 사용하여 인식률을 향상시키고 속도를 개선하는 오프라인 필기체 숫자 인식 시스템을 제안한다. 전처리 과정에서 세선화를 수행하여 그 결과로 나온 이미지에서 획을 추출하여 각 획의 방향코드를 생성한 후 이를 인식기의 입력으로 사용한다.

본 논문은 다음과 같이 모두 5장으로 구성되어 있다. 2장에서는 관련 연구로 SOG*(Improved Self-Organized feature Graph) 세선화를 소개하고 3장에서는 각 획의 방향 코드를 추출하는 방법을 기술한다. 4장에서는 실험 및 실험 결과를 분석하고 마지막 5장에서는 결론을 맺는다.

2. 관련 연구(SOG* 세선화)

SOG* 세선화 방법은 세선화를 위한 학습 과정에서 경쟁층의 뉴런을 재정돈 하는 과정을 거쳐 승자 뉴런을 선택할 때 모든 뉴런을 한번에 모두 재정돈 하는 것이 아니라 재정돈 할 뉴런의 수를 단계별로 증가 시키며 학습시키는 점층적 학습 알고리즘과 학습 과정에서 뉴런들이 각 입력 패턴의 구조를 나타내는 위치를 찾아 안정화되는 시점을 파악하여 이후 학습에서 기존 정렬을 그대로 사용하는 SOG의 변형 방법을 사용한 SOG 기반의 고속 세선화 알고리즘이다[3].

다음 그림 1에서 SOG* 세선화의 결과를 보여 주고 있으며 각 세선화 결과는 원 이미지의 특성을 유지하는 우수한 결과를 나타내었고 수행 속도 또한 기존 방법보다도 우수한 $O((\log M)^3)$ 을 나타내었다. 세선화 과정에서의 경쟁층 뉴런의 수는 17개로 고정한다[3].



(a) 원 이미지 (b) SOG* 결과

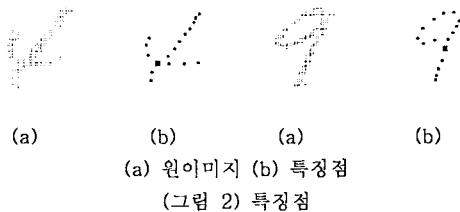
(그림 1) 세선화 결과

3. 방향 코드의 추출

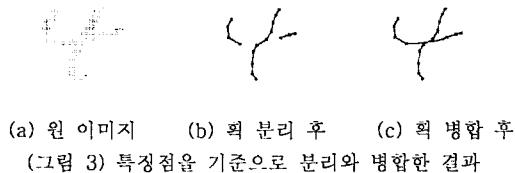
3.1. 특징 점의 추출과 획 분리와 병합

세션화의 결과로 나온 17개의 점에서 각 점과 점 사이의 거리와 방향, 원 이미지와 세션화 이미지와의 연관성으로 특징 점을 추출하고 동시에 획의 분리를 하며, 마지막으로 획 병합을 이룬다. 여기서 특징점이라는 것은 어떤 점에서 둘 이상의 방향으로 분기 가능한 점을 특징점으로 정의하여 이 점에서 획을 분리한다.

먼저 세션화 결과 이미지에서 시작점을 선택하여 정렬을 수행하는데, 시작점은 세션화 이미지의 절단면을 2차원 좌표로 생각하여 원점에서 가장 가까운 점을 시작점으로 선택한다. 다음으로 정렬을 수행하는데 시작점을 기준으로 각 점에 순서를 부여하는데 시작점은 가장 가까운 점을 다음 점으로 결정하고 그 이외는 점과 점 사이의 거리가 임계치 이하이면 후보로 두고, 이전 점과의 기울기의 차를 고려한다. 또한 원 이미지와 연관성을 조사하여 후보들 중에서 가장 적절한 점을 다음 점으로 선택하게 된다. 이때 적절한 점이 둘 이상이 나오는 경우 이를 특징점으로 판단하게 된다. 특징점은 다음 그림 2에서 다른 점들보다 굵게 표시된 점이다.



위 그림 2와 같이 특징 점을 기준으로 획을 분리하게 되면 다음 그림 3 (a)와 같이 분리되게 된다. 이렇게 분리된 획들은 다시 병합 과정을 거쳐 최종적으로 획들로 구성하게 된다. 병합 과정을 거치는 이유는 하나의 획임에도 불구하고 불필요하게 분리되어 둘 이상의 획으로 나타나는 경우가 생기기 때문이다.



위 그림 3 (b)와 같이 가로획이 하나로 이어질 수 있음에도 불구하고 특징점을 기준으로 분리되어 있다. 불필요하게 분리된 획들을 병합함으로써 단순한 획을 생성해 낼 수 있다. 그림 3의 (c)에서는 획 병합 과정을 거친 후의 이미지를 보여주고 있다.

3.2. 획의 방향 코드 생성

3.1의 각 단계를 거친 후에 전처리의 마지막 단계로 방향 코드를 추출하게 되는데 각 획들로 정렬된 순서에 따라 8개의 절대 방향으로 코드를 생성하게 된다[4,5]. 방향 코드는 각 획별로 생성하게 되며 반복되는 코드는 하나로 줄임으로써 정규화를 시키게 된다. 이렇게 생성된 방향 코드는 각 획별로 최대 12개가 생성되었다. 생성된 방향 코드로 획을 특징 지을수가 있으며 생성된 획들은 모두 15개의 패턴으로 분류되었다. 이 코드는 획 인식기의 입력으로 들어가게 된다.

3.3. 획 인식기

인식기는 각 획 당 은닉층을 갖는 3층 퍼셉트론 구조로 구성되었으며 입력은 96개(12개 코드 \times 8방향), 출력은 15개(15개의 획 패턴), 은닉층은 실험적으로 42개로 결정하였으며 학습 단계에서는 학습률 0.02, 임계치 0.9, 인식 단계에서는 학습률 0.02, 임계치 0.9로 결정하였다.

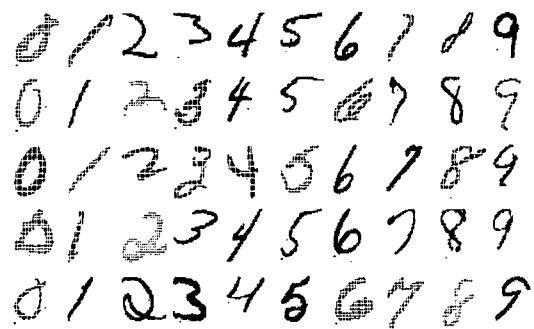
각 획별로 인식기를 통해 인식한 획들은 최종적으로 각 획들의 조합으로써 숫자를 인식하게 된다. 숫자 인식은 인식기에서 출력된 획들과 특징점을 기준으로 결정된 공간 정보를 이용하게 된다. 특징점을 기준으로 각 획들이 분리되기 때문에 각 획들이 특징점의 어느 위치에 존재하는지를 정보로써 저장하여 최종 숫자 인식시에 사용한다.

4. 실험 및 실험 결과

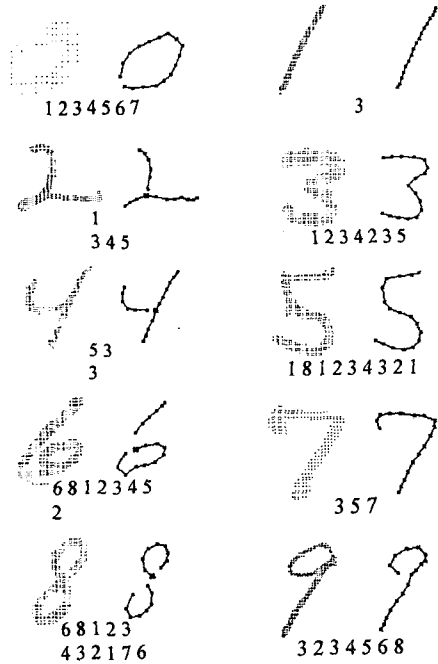
실험은 펜티엄 III 1GHz, Windows 2000, VC++ 6.0의 환경에서 실험하였으며 실험에 사용된 숫자 이미지들은 Concordia 대학교의 필기체 숫자 데이터베이스이다.

본 논문의 실험은 숫자 샘플의 일부를 학습에 사용하고 일부를 인식 테스트하는 기존의 방법[5,6]과 달리 필기체 숫자가 지니고 있는 특성을 미리 파악하여 패턴을 저장하여 학습에 사용하고 모든 샘플을 테스트하는 방식을 사용하였다.

다음 그림 4에서는 실험에 사용된 숫자 이미지들을 보여주고 있으며 그림 5에서는 실험 결과로 나온 각 획들과 획들의 코드 값을 보여 주고 있다.

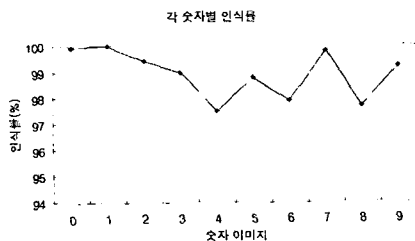


(그림 4) 실험에 사용된 Concordia 대학의 숫자 이미지



(그림 5) 실험 결과로 나온 획과 방향 코드

본 실험에서 Concordia 대학교의 필기체 숫자 데이터베이스의 6,000개 데이터로 인식률을 테스트 한 결과 98.89%의 정 인식률과 0.4%의 오인식률, 0.71%의 미인식률을 나타내었고, KAIST 오프라인 필기체 숫자 데이터베이스에 대해서는 각각 99.25%, 0.2%, 0.55%, 연세대학교 숫자 데이터베이스에 대해서는 각각 98.85%, 0.4%, 0.71%의 인식률을 나타내었다. Concordia 대학교의 데이터베이스에 대해 실험을 한 다른 논문들과 비교해 보면 임길택[7] 98%, 류강수[8] 98.2%, 김현돈[9] 94.45%의 인식률로 본 논문의 인식률이 우수함을 알 수 있다.



(그림 5) 각 숫자별 인식률

그림 5에서는 각 숫자별로 인식률을 나타내고 있는데, 획이 단순한 숫자 0, 1, 5, 7 등은 다른 숫자에 비해 인식률이 높게 나타난다. 다른 숫자들도 비교적 높은 인식률을 나타내지만 0, 1, 5, 7 보다 낮은 이유는 패턴의 다양함과 획의 유사 패턴의 발생이 주된 원인이었다.

5. 결론

본 논문에서는 오프라인 필기체 숫자인식을 위하여 단순한 전처리로 세선화와 방향 코드를 사용하였다. 빠르고 결과가 우수한 SOG* 세선화 기법을 이용하여 세선화를 수행하고, 세선화의 결과로 나온 17개의 점으로 획을 분리, 병합한 후 방향 코드를 추출하여 획 인식기의 입력으로 들어가게 된다. 생성된 획과 공간 정보를 이용하여 최종적으로 숫자를 인식하는 방식이다.

복잡한 전처리의 과정을 아주 단순화 시킴으로써 특징 추출에 소요되는 시간을 줄일 수 있었고, 이미지의 크기나 기울기에 제약을 없었다. 또한 미리 생성해 놓은 코드 패턴들을 학습에 사용하였으므로 특정 데이터베이스의 일부를 학습하고 일부를 테스트하는 방법이 아닌 한 번의 학습으로 모든 이미지를 인식에 사용할 수 있다.

본 논문은 타 논문과 유사한 높은 인식률을 가지며 전처리 작업의 단순화를 이루어 효율성이 높음을 알 수 있다.

참 고 문 헌

- [1] 방국준, 조남신, 강창언, 홍대식, "필기체 숫자 인식을 위한 병렬 자구성 계층 신경 회로망", 전자공학회논문지, 제 33 권 B편 제 7호, pp.1193-1202, 1996
- [2] 이성환, "패턴 인식의 원리 I, II", 홍릉과학출판사, 1997
- [3] 이찬희, 정순호, "개선된 SOG 기반 고속 세선화 알고리즘 (SOG)", 한국정보처리학회논문지, 제 8-B권, 제 6호, pp.651-656, 2001
- [4] H. Freeman, "On the Encoding of Arbitrary Geometric Configuration", IEEE Trans. on Electronic Computers, Vol. EC-10, pp.260-268, June 1961.
- [5] 김백섭, 송해정, "필기체 문자 인식을 위한 체인코드에 기반한 효율적인 전처리 및 특징 추출", 한국정보과학회논문지 (B), 제 25권, 제 12호, pp.1788-1796, 1998.
- [6] 박창순, 김두영, "오프라인 필기체 숫자 인식을 위한 다양한 특징들의 성능 비교 및 인식을 개선 방안", 한국정보처리학회논문지, 제 3권, 제 4호, pp.915-925, 1996
- [7] 임길택, 남윤석, 진성일, "회전 및 이동 영상을 이용하는 모듈 구조 신경망 기반 필기체 숫자 인식", 한국정보처리학회논문지, 제 7권, 제 6호, pp.1834-1843, 2000.
- [8] 류강수, 김우태, 진성일, "다중 특징과 모듈화된 신경회로 망을 이용한 인쇄체 및 필기체 혼용 숫자 인식", 전자공학회 논문지, 제 32권 B편, 제 10호, pp.1347-1357, 1995.
- [9] 김현돈, 조성배, "유전자 알고리즘을 사용한 구조적응 자기 구성 지도의 최적화", 퍼지및지능시스템학회논문지, Vol. 11, No. 3, pp.223-230, 2001.