

체인 코드 트리밍과 가중 원형 정합을 이용한 인쇄체 숫자 인식

정민철

상명대학교 컴퓨터시스템공학과
(mjung@smu.ac.kr)

.....

본 논문에서는 인쇄체 숫자를 인식하기 위해 가중 원형 정합(weighted template matching) 방법을 제안한다. 원형 정합은 입력 영상 전체를 하나의 전역적인 특징으로 처리하는데 반해, 제안된 가중 원형 정합은 패턴의 특징이 나타나는 국부적인 영역에 해밍 거리(Hamming distance)의 가중치를 두어 패턴 특징을 강조하여 숫자 패턴의 인식률을 높인다. 또한 원형 정합은 영상을 이진화(binazation)할 때 발생하는 패턴의 윤곽선을 울퉁불퉁하게 만드는 랜덤 노이즈에 취약하다. 이러한 랜덤 노이즈를 제거하기 위해 본 논문에서는 체인 코드 트리밍(chain code trimming)방법을 제안한다. 체인 코드 트리밍은 패턴의 외부 윤곽선과 내부 윤곽선의 체인 코드 분석을 통해 특정 유형의 체인 코드를 제거, 교정함으로써 이루어진다. 실험에서는 기존의 원형 정합을 사용했을 때와 체인 코드 트리밍을 한 후 가중 원형 정합을 사용했을 때의 혼돈 행렬(confusion matrix)을 서로 비교한다. 실험 결과는 본 논문에서 제안한 방법에 의해 인쇄체 숫자의 인식률이 크게 향상된 것을 보인다.

.....

논문접수일 : 2007년 04월 게재확정일 : 2007년 10월 교신저자 : 정민철

1. 서론

문자 인식(OCR ; Optical Character Recognition)은 영상으로 된 문서 이미지를 컴퓨터가 처리할 수 있는 문자 코드로 바꾸어 문서내의 문자 정보를 처리하는 것이다[1]. 그 응용 분야는 기존 서류들의 자동 입력으로부터 신용카드 전표 인식, 팩스 문서 자동 인식에 이르기까지 매우 광범위하

다. 문자 인식 시스템에 대한 초기 연구는 문자 영상을 컴퓨터에 입력하는 방법과 컴퓨터의 능력에 오히려 제한을 받아 그 발전이 늦어졌다. 현재 하드웨어적인 기술상의 문제는 극복되었으나 소프트웨어적인 기술상의 문제는 아직 완전히 해결되지 않았다. 문자 인식 분야에는 영문자 인식, 한글 인식, 한자 인식, 숫자 인식 등 여러 가지가 있는데 이 중 특히 인쇄체 숫자 인식은 자동차 번호판

인식, 수표, 지로(GIRO) 등의 정보 처리에 활용 범위가 광범위하다[2-5]. 인쇄체 숫자는 내부 구조적으로 서로 비슷한 형태로 패턴이 구성되어, 열화된 이진 영상에서는 특징 추출이 어려우며 다른 문자 인식과는 달리 숫자 인식에서는 문맥(context)이나 사전 참조 같은 후처리(postprocessing) 과정도 사용할 수 없다. 또한 숫자 인식은 하나의 오류가 있다 하더라도 결과 처리에 있어 치명적이다. 숫자 인식에 관한 연구로는 크게 통계적 패턴 인식과 인공 신경망을 이용한 인식으로 나눌 수 있다. 통계적인 방법으로서 선형 판별함수(LDF), 이차형 판별함수(QDF), 정칙화 판별함수(RDF) 등의 모수적인 방법과, k-최근접 이웃방법(k-NN) 등의 비모수적인 방법을, 인공 신경망 방법으로는 다층 퍼셉트론(MLP)을 선택 비교하였을 때, 다층 퍼셉트론이 가장 높은 인식률을 보였다[6]. 최근의 연구 중 하나인 프랙탈 차원과 수정된 에논 어트랙터를 이용한 연구는 패턴의 미세한 차이를 고감도로 식별할 수 있는 카오스 이론의 프랙탈 차원과 에논 시스템에서 발생하는 스트레인지 어트랙터를 이용하여 특징을 추출하여 혼동 숫자를 정량적으로 분석하였다[7]. 그러나 현재까지도 인쇄체 숫자의 오인식률은 여전히 존재한다. 이는 인쇄체 숫자는 기하학적으로 구조가 단순하고 비슷하여, 기존의 문자 인식에서 사용하는 전역적 특징 추출법이 인쇄체 숫자 인식에는 적용되기 어려운 경우가 많기 때문이다. 따라서 숫자의 기하학적 구조보다는 획의 굵기 변화 같은 미묘한 차이를 특징으로 추출해야 하는데 각각의 픽셀을 특징으로 하는 원형 정합(template matching) 방법이 간단하면서도 신속하게 숫자 인식을 할 수 있다[8-10]. 원형 정합 방법은 패턴의 회전과 크기의 변화가 적을 경우 효과적이다[11, 12]. 그러나 패턴의 회전과 크기의 변화가 적은 경우에도, 원형 정합 방법

은 패턴의 모든 픽셀을 동일한 가중치로 취급하므로 인쇄체 숫자와 같이 특징이 국부적으로 존재하는 패턴에는 효율적이지 못하다. 이를 개선하기 위해 본 논문에서는 숫자 패턴의 특징이 존재하는 부분에 국부적으로 가중치를 두어 정합하는 가중 원형 정합 방법을 제안한다. 또한 패턴의 크기 정규화(size normalization) 과정에서 첨부되는 랜덤 노이즈(random noise)를 제거하기 위한 방법으로 체인 코드 트리밍(chain code trimming)을 제시하여 입력 숫자 패턴의 크기 변화에도 높은 인식률을 유지할 수 있도록 한다.

2. 원형 정합

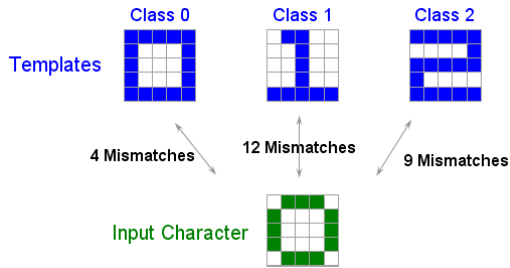
원형 정합은 입력 영상에서 특징을 추출하는 과정 없이 입력 영상 전체를 하나의 전역적인 특징으로 처리한다. 즉, 입력되는 패턴은 각각의 클래스를 나타내는 원형(template) 또는 프로토타입(prototype)과 직접적으로 비교된다. 분류는 최대의 일치도 또는 최소의 불일치도에 따라 이루어진다. 수학적으로 비교는 거리의 측정을 의미한다. 입력 패턴 x 와 클래스 C_i 의 프로토타입 사이의 거리는 함수 $d(x, C_i)$ 에 의해 계산되고 x 는 이 함수를 최소화하는 클래스에 할당된다. 본 논문에서는 이를 위해 해밍 거리(Hamming distance)를 사용했다. 해밍 거리는 두 패턴의 같은 위치상의 부울리안(Boolean) 벡터를 서로 비교하여 동일하면 0을 다르면 1을 합계한다. 해밍 거리의 정의는 sum mod 2의 연산을 필요로 하는데 기호, \oplus 로 나타내면 :

$$0 \oplus 0 = 0; 1 \oplus 1 = 0; 0 \oplus 1 = 1; 1 \oplus 0 = 1$$

따라서 해밍 거리는 다음과 같이 나타낼 수 있다[13].

$$D_h[k, l] = \sum_{i=1}^n (x_{ik} \oplus x_{il}) \quad (1)$$

해밍 거리가 적을수록 두 패턴은 유사하다고 할 수 있다. [그림 1]은 해밍 거리를 이용한 원형 정합의 예를 보인다. 비교되는 패턴의 모든 픽셀에 대해 일치될 때 0이 불일치될 때 1이 합산된다. 그림에서는 첫 번째 원형이 최소 불일치로 입력 글자에 대한 결과로 선택된다.



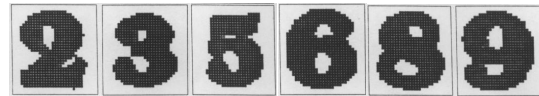
[그림 1] 해밍 거리를 이용한 원형 정합

원형 정합은 분류할 클래스의 수가 많으면 시간이 많이 걸리고, 입력 패턴의 변형에 취약하다는 단점이 있다. 그러나 패턴의 내부 구조가 서로 비슷하여 특징 추출이 어렵고 분류되는 클래스의 수가 적은 인쇄체 숫자 인식 등에는 효과적으로 적용되어 사용될 수 있다.

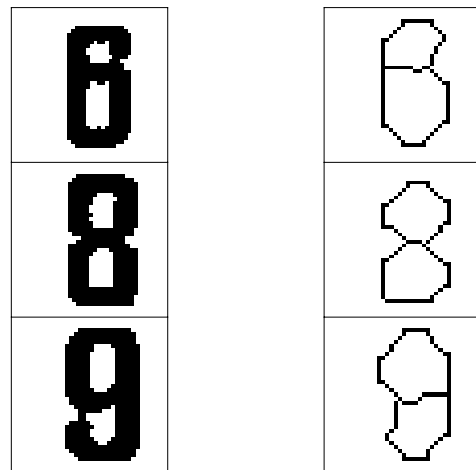
3. 혼돈 숫자와 골격화

인쇄된 문자 패턴을 스캔하여 영상 이진화(binazation)하면, 우리가 종이에서 보는 인쇄된 문자와는 전혀 다른 패턴 영상이 형성된다. 이는 스캔할 때 랜덤 노이즈가 첨가되거나 영상 이진화 자체 알고리즘의 문제로 이진화 영상을 입력으로 하는 OCR 시스템에게는 불가피한 현상이다. 숫자

는 문자 내부의 구조가 단순하다. 단순한 만큼 몇몇 문자는 스캔되고 영상 이진화한 후에는 문자의 내부 구조가 비슷해 OCR 시스템에게는 혼돈을 일으켜 오류를 발생시킨다. 폰트에 따라 정도의 차이는 있으나 '5와 6', '6, 8과 9'는 문자 내부적 구조가 유사하다. [그림 2]는 이러한 혼돈을 주는 숫자들의 예를 보여준다. OCR 시스템에서는 문자의 구조를 알아내기 위해 골격화(skeletonization) 처리를 사용한다. 문자의 골격은 문자에 내재된 구조를 설명하기 때문에 중요한 측정치가 된다. 보통 이러한 구조를 측정하고 나서 이미 알려진 구조와 이 구조를 비교하여 문자의 특성을 알아내 문자를 인식할 수 있다. 그러나 숫자의 구조적 유사함은 [그림 3]과 같이 문자 내의 획이 서로 접합될 경우 골격화 같은 기하학적인 특징 추출은 무의미해진다[14, 15].



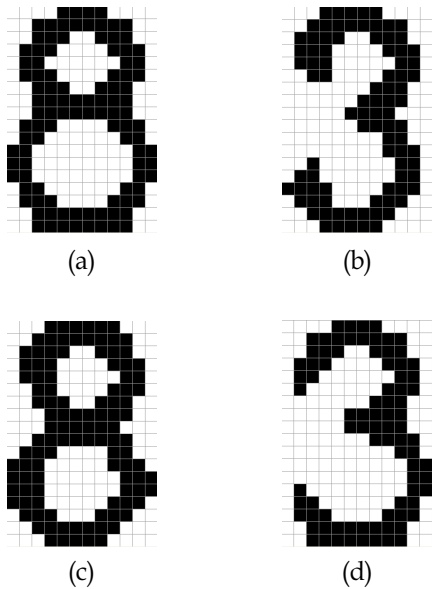
[그림 2] 혼돈 숫자 : 위의 모든 숫자는 내부 구조적으로 두 개의 홀(hole)을 가진다.



[그림 3] 골격화의 한계 : 좌측의 영상을 골격화한 우측의 영상은 구조적으로 완전히 동일하다.

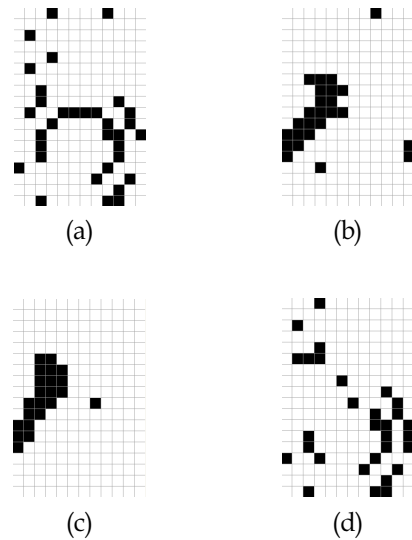
4. 가중 원형 정합

원형 정합은 입력 영상 전체를 하나의 전역적인 특징으로 한다. 따라서 입력 영상과 원형의 모든 픽셀에 대해 일률적으로 해밍 거리를 취한다. 그러나 이 경우 패턴의 주요한 특징을 나타내는 부분의 해밍 거리가 그렇지 못한 부분의 해밍 거리와 총합으로 합해지고, 그 총합은 패턴 고유의 특징을 나타내지 못한다. 또한 단순한 해밍 거리의 사용은 [그림 4]에서 보는 바와 같이 해밍 거리 오류를 발생시킨다. [그림 4(a)]와 [그림 4(b)]가 원형일 때, 미지의 패턴인 [그림 4(c)]와 [그림 4(d)]가 입력된다고 가정한다. 여기서 보이는 [그림 4(b)]와 [그림 4(d)]의 패턴 '3'을 참고 문헌 [13]의 예에서는 바운딩 박스(bounding box)처리와 크기 정규화(size normalization)를 하지 않았으나, 원형 정합의 예를 들기에는 큰 무리는 없다.



[그림 4] 해밍 거리 오류 : (a)와 (b)는 프로토타입 문자, (c)와 (d)는 프린트한 후 스캔된 이미지([13]에서 참조).

[그림 4]에서 보면, [그림 4(c)]의 '8'과 [그림 4(a)]의 '8'사이의 해밍 거리 D_h 는 33인 반면 [그림 4(c)]의 '8'과 [그림 4(b)]의 '3'사이의 해밍 거리 D_h 는 25이다. 마찬가지로 [그림 4(d)]의 '3'과 [그림 4(a)]의 '8' 사이의 해밍 거리 D_h 는 22인 반면, [그림 4(d)]의 '3'과 [그림 4(b)]의 '3'사이의 해밍 거리 D_h 는 30이다. 따라서 미지의 패턴인 [그림 4(c)]는 해밍거리가 보다 적은 3으로 분류되고, 미지의 패턴인 [그림 4(d)]는 해밍거리가 보다 적은 8로 분류된다. [그림 5]는 [그림 4]에서 보인 패턴의 영상 차분을 나타낸다. 그림에서 보는 바와 같이 같은 클래스 패턴의 경우 영상의 차가 전역적으로 분산되어 나타나는 데 반해, 다른 클래스 패턴의 경우 영상의 차가 국부적으로 집중되어 나타난다.



[그림 5] 해밍 거리 : (a) [그림 4(c)]의 '8'과 [그림 4(a)]의 '8'사이의 해밍 거리 $D_h = 33$, (b) [그림 4(c)]의 '8'과 [그림 4(b)]의 '3'사이의 해밍 거리 $D_h = 25$, (c) [그림 4(d)]의 '3'과 [그림 4(a)]의 '8' 사이의 해밍 거리 $D_h = 22$, (d) [그림 4(d)]의 '3'과 [그림 4(b)]의 '3'사이의 해밍 거리 $D_h = 30$.

본 논문에서 제안한 가중 원형 정합은 패턴의 특징이 나타나는 국부 영역에 해밍 거리의 가중치를 두어 패턴의 국부적 특징을 강조하여 패턴의 인식률을 높이는 것이다. 즉, 가중 원형 정합은 다음과 같이 나타낼 수 있다. 여기서 d_1 과 d_2 사이의 패턴의 고유 특징을 중점적으로 나타내는 국부적 영역이다. 이곳에 w 의 가중치를 준다.

$$D_{wh}[k, l] = \sum_{i=1}^{d_1-1} (x_{ik} \oplus x_{il}) + w \sum_{i=d_1}^{d_2} (x_{ik} \oplus x_{il}) + \sum_{i=d_2+1}^n (x_{ik} \oplus x_{il}) \quad (2)$$

인식에서 혼동을 발생시키는 인쇄체 숫자는 주로 패턴의 중간 부분에 국부적인 패턴의 특징이 존재한다. 따라서 패턴의 중간 부분에 가중치를 두어 가중 해밍 거리를 구하면 혼동되는 인쇄체 숫자의 인식률을 높일 수 있다.

를 줄일 수 있다. 앞에서 보인 [그림 4]에 본 논문에서 제안한 가중 원형 정합을 적용한 예를 들기 위해 식 (2)에 $d_1 = 73$, $d_2 = 144$ 와 가중치 $w = 6$ 을 대입한다. d_1 에서 d_2 사이의 패턴의 중간 부분을 나타내며 해밍 거리가 전체적으로 클 때 가중치도 크게 해야 한다. 이때 [그림 4(c)]의 '8'과 [그림 4(a)]의 '8' 사이의 가중 해밍 거리 D_{wh} 는 113이고, [그림 4(c)]의 '8'과 [그림 4(b)]의 '3' 사이의 가중 해밍 거리 D_{wh} 는 115이다. 마찬가지로 [그림 4(d)]의 '3'과 [그림 4(a)]의 '8' 사이의 가중 해밍 거리 D_{wh} 는 107이고, [그림 4(d)]의 '3'과 [그림 4(b)]의 '3'사이의 가중 해밍 거리 D_{wh} 는 75이다. 따라서 본 논문에서 제안한 가중 원형 정합 방법을 이용하면 미지의 패턴인 [그림 4(c)]는 가중 해

밍거리가 보다 적은 8로, 미지의 패턴인 [그림 4(d)]는 해밍거리가 보다 적은 3으로 성공적으로 분류될 수 있다.

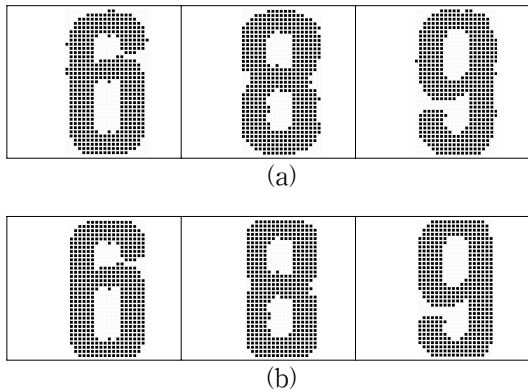
5. 체인 코드 트리밍(chain code trimming)을 사용한 랜덤 노이즈 제거

영상을 스캐닝하고 이진화할 때 패턴에 한 개나 두개의 픽셀이 첨부되는 랜덤 노이즈는 [그림 6(a)]에서 보듯이 패턴의 윤곽선을 울퉁불퉁하게 만든다. 이러한 패턴 윤곽선의 랜덤 노이즈는 입력 영상을 크기 정규화(size normalization)할 때 패턴의 내부 구조를 변형시키는 원인이 되기도 한다. [그림 2]에서 보였던 문자 내부의 획이 접합된 경우에도 이러한 패턴 윤곽선의 랜덤 노이즈가 원인인 경우가 대부분이다. 또한 이러한 패턴 윤곽선의 랜덤 노이즈는 원형 정합을 사용할 때 영상의 전체 픽셀을 랜덤 노이즈만큼 자리 이동 시켜 두 패턴의 같은 위치상의 부울리안 벡터를 서로 비교하는 해밍 거리의 오차를 크게 하여 신뢰도를 저하시킨다. 가장 간단하게 트리밍하는 방법은 입력 패턴의 상하 좌우에서 한 픽셀 라인씩 단순히 깎아 내는 것이다. 그러나 이 방법은 [그림 6(b)]에서 보듯이 패턴의 테두리에 존재하던 랜덤 노이즈는 제거하지만 여전히 패턴의 내부에 존재하는 랜덤 노이즈는 제거하지 못한다. 패턴의 윤곽선에 첨부된 랜덤 노이즈를 제거하기 위해 본 논문에서는 체인 코드 트리밍 방법을 제안한다.

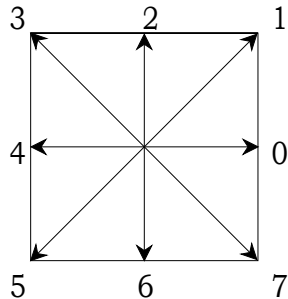
5.1 체인 코드

체인 코드(chain code)는 패턴에서 흰 픽셀과 검은 픽셀이 만나는 부분인 테두리(edge)를 시계 방향이나 반시계 방향으로 따라가면서 테두리 픽

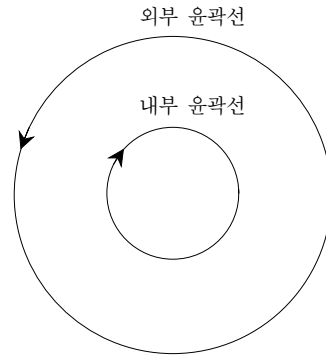
셀의 이동 방향을 0에서 7로 나타낸다[16]. [그림 7]은 체인 코드 방향의 정의를 보인다. 체인 코드의 방향은 테두리 픽셀의 이동 방향으로 나타내는데, 0°는 0으로, 45°는 1로, 90°는 2로, 135°는 3으로, 180°는 4로, 225°는 5로, 270°는 6으로, 315°는 7로 각각 정의한다. [그림 8]은 외부 윤곽선과 내부 윤곽선의 체인 코드의 방향을 나타낸다. 패턴 외부 윤곽선은 그림처럼 반시계 방향으로 따라가면서 체인 코딩을 하고, 패턴 내부 윤곽선은 그림처럼 시계 방향으로 따라가면서 체인 코딩을 한다. 체인 코딩은 두 경우 모두 패턴 윤곽선의 최상위 좌측에서 시작한다.



[그림 6] 트리밍(trimming)을 사용한 패턴 윤곽의 랜덤 노이즈 제거
(a) 트리밍 전의 패턴과 (b) 단순 트리밍 후의 패턴



[그림 7] 체인 코드의 방향성 할당











[그림 8] 외부 윤곽선과 내부 윤곽선의 체인 코드의 방향









5.2 랜덤 노이즈의 체인 코드와 교정

패턴 외부 윤곽선을 반시계 방향으로 체인 코딩을 하면서 체인 코드 리스트가 특정한 값, 즉 랜덤 노이즈의 체인 코드를 가지면 그 값을 지정한 값으로 변경한다. [그림 9]에서 보듯이 패턴의 외부 윤곽선은 4가지 경우의 한개 픽셀 랜덤 노이즈를 가질 수 있다. 패턴 외부 윤곽선의 체인 코드가 이러한 랜덤 노이즈의 체인코드 즉, '35', '57', '71', '13'이면 그림에서 보인바와 같이 이에 해당되는 체인 코드로 대체한다. 대체되는 체인 코드는 '44', '66', '00', '22'와 같이 한개 픽셀이 외부 윤곽선에 되어 나오는 것을 반듯한 윤곽선으로 교정한다. 변경된 체인 코드 리스트로 윤곽선 영상을 다시 그리면 한개 픽셀의 랜덤 노이즈가 사라진, 즉 체인 코드 트리밍된 영상이 된다. 또한 [그림 10]에서 보듯이 패턴의 외부 윤곽선은 4가지 경우의 두개 픽셀 랜덤 노이즈를 가질 수 있다. 패턴 외부 윤곽선의 체인 코드가 이러한 랜덤 노이즈의 체인 코드 즉, '345', '567', '701', '123'이면 그림에서 보인바와 같이 이에 해당되는 체인 코드로 대체한다. 마찬가지로 패턴 내부 윤곽선의 경우에도 시계 방향으로 따라가면서 랜덤 노이즈 체인 코드(한 개

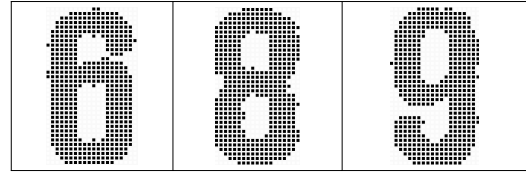
픽셀 랜덤 노이즈의 경우 '17', '31', '53', '75'를 같은 방식으로 교정한다. [그림 11]은 제안된 체인 코드 트리밍에 의해 테두리 랜덤 노이즈가 제거된 영상을 보인다.

랜덤 노이즈 제거 전		랜덤 노이즈 제거 후	
영상	체인 코드	영상	체인 코드
	35		44
	57		66
	71		00
	13		22

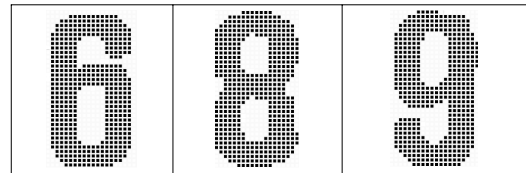
[그림 9] 패턴의 외부 윤곽선이 한 개 픽셀의 랜덤 노이즈를 가지는 경우

랜덤 노이즈 제거 전		랜덤 노이즈 제거 후	
영상	체인 코드	영상	체인 코드
	345		444
	567		666
	701		000
	123		222

[그림 10] 패턴의 외부 윤곽선이 두 개 픽셀의 랜덤 노이즈를 가지는 경우



(a)

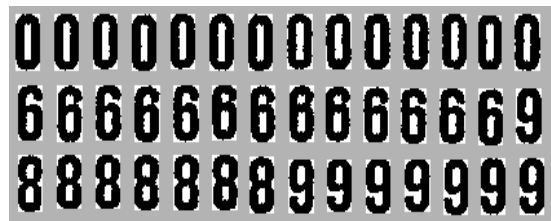


(b)

[그림 11] 체인 코드 트리밍을 사용한 패턴 테두리의 랜덤 노이즈 제거 (a) 트리밍 전의 원래 패턴, (b) 체인 코드 트리밍 후의 패턴

6. 실험 결과

[그림 12]은 본 연구에서 사용된 숫자 영상의 예를 보여준다. 이러한 숫자 영상을 각각 100개씩 입력으로 하여 단순 원형 정합과 본 논문에서 제안한 가중 원형 정합을 각각 사용하여 두 방법을 비교 실험하였다.



[그림 12] 실험에 사용된 숫자 영상의 예

<표 1>은 트리밍을 하지 않고 단순 원형 정합을 사용했을 때의 혼돈 행렬(confusion matrix)을 나타낸다. <표 2>는 체인 코드 트리밍을 한 후 가중 원형 정합을 사용했을 때의 혼돈 행렬이다. 표

에서 보듯이 본 논문에서 제안한 방법에 의해 인쇄체 숫자 인식률이 크게 향상된 것을 알 수 있다. 실험에서 거부율(reject)은 두 경우 모두 0로 하여 해밍 거리의 신뢰도가 아주 낮은 경우에도 결과를 출력하게 하였다. 따라서 각 표에서 해밍 거리의 신뢰도가 낮은 경우에는 오류 발생률이 높아졌다.

<표 1> 단순 원형 정합을 사용했을 때의 혼돈 행렬(confusion matrix)

출력 입력	0	2	3	5	6	8	9	오류(error)
0	93	0	0	0	5	2	0	7
2	0	98	1	0	0	0	1	2
3	0	0	95	0	0	3	2	5
5	0	0	0	98	2	0	0	2
6	2	0	0	0	91	7	0	9
8	0	0	0	0	4	96	0	4
9	2	0	0	0	3	1	94	6

<표 2> 가중 원형 정합을 사용했을 때의 혼돈 행렬(confusion matrix)

출력 입력	0	2	3	5	6	8	9	오류(error)
0	100	0	0	0	0	0	0	0
2	0	100	0	0	0	0	0	0
3	0	0	99	0	0	1	0	1
5	0	0	0	100	0	0	0	0
6	0	0	0	0	98	2	0	2
8	0	0	0	0	1	99	0	1
9	0	0	0	0	0	0	100	0

7. 결론

본 논문에서는 원형 정합 방법을 개선한 가중 원형 정합 방법을 제안하였다. 인쇄체 숫자 중 몇

개는 문자 내부 구조가 유사하여 스캐닝되고 영상 이진화되면서 랜덤 노이즈가 첨부되면 문자의 특징 추출을 어렵게 한다. 더구나 문자의 구조를 알아내기 위해 골격화 처리를 사용하는 OCR 시스템은 이러한 숫자를 서로 혼돈하여 오류를 발생시킨다. 따라서 숫자의 기하학적 구조보다는 획의 굵기 변화 같은 미묘한 차이를 특징으로 추출해야 하는데 각각의 픽셀을 특징으로 하는 원형 정합 방법이 간단하면서도 신속하게 숫자 인식을 할 수 있다. 그러나 원형 정합 방법은 모든 픽셀을 동일하게 취급함으로써 패턴 고유의 특징을 나타내지 못한다. 본 논문에서 제안된 가중 원형 정합 방법은 패턴의 특징이 나타나는 영역에 해밍 거리의 가중치를 두어 패턴 특징을 강조하여 패턴의 구별을 확실히 하며, 이는 실험 결과에서 검증되었다. 또한 원형 정합 방법이 랜덤 노이즈에 따라 전체 패턴의 픽셀이 어긋나는 것을 방지하기 위해 체인 코드 트리밍을 사용하여 랜덤 노이즈를 제거하였다. 원형 정합 방법의 한계이자 제안된 방법의 한계로서는 원형으로 저장된 프로토타입의 폰트를 벗어난 다양한 폰트의 입력에 대해 인식률이 저하된다는 것이다. 사실 다양한 폰트로 쓰인 문서의 문자 인식에서 높은 인식률을 계속 유지하는 OCR 시스템을 만드는 것은 아직 풀어야 할 연구 과제이다. 본 연구에서는 각각 다른 폰트에 대해 다른 프로토타입을 만들어 이 문제를 해결한다. 그러나 여전히 몇몇 알려진 폰트에 대해서 이 방법이 가능하며 새로운 폰트에 대해 새로운 프로토타입을 만들어야 하는 단점이 있다.

참고문헌

- [1] G. Nagy and S. Seth, "Modern Optical Character Recognition", The Froehlich/Kent

- Encyclopedia of telecommunications, F. Froehlich and A. Kent, eds., Marcel Dekker, Inc., Vol. 11(1996), 473~531.
- [2] 김진숙, 변영철, 김경환, 최영우, 이일병, “지로 서식 문서의 인쇄체 숫자 인식”, 한국정보과학회 가을학술발표논문집, 제26권, 제2호(1999), 446~448.
- [3] 류강수, 김경훈, 진성일, “지로 OCR에서 인쇄체 숫자 영역 분리 및 신경망 기반의 인식에 대한 연구”, 대한전자공학회 학술대회 논문집, 제6권, 제1호(1993), 253~257.
- [4] 서동훈, 정해권, 이원돈, “수직, 수평 성분을 이용한 한국 자동차 번호판 인식”, 한국정보과학회 가을 학술발표논문집, 제30권, 제2호(2003), 610~612.
- [5] 이득용, 오일석, “숫자 인식에 기반한 자동차 번호판 추출”, 한국 컴퓨터종합학술대회 논문집, Vol.34, No.1(2007), 407~411.
- [6] 정선화, 김수형, 조완현, “통계적 패턴인식과 신경망을 이용한 필기 숫자 인식방법의 성능 비교”, 한국정보과학회 1998년도 봄 학술발표논문집, 제25권, 제1B호(1998), 719~721.
- [7] 손영우, “프랙탈 차원과 수정된 예능 어트랙터를 이용한 인쇄체 숫자인식”, 한국멀티미디어학회 논문지, 제6권, 제1호(2003), 89~96.
- [8] 백순흠, 박종원, “선분 추출과 원형 정합을 이용한 여러 인쇄 활자체의 영문자와 숫자의 인식”, 정보과학회논문지, 제19권, 제2호(1992), 123~132.
- [9] 박덕원, 박종원, “3×3 템플릿을 이용한 여러 영문 활자체의 인식”, 정보과학회논문지B, 제23권, 제6호(1996), 625~634.
- [10] 임길택, “문자형식 분류 기반의 인쇄체 문자 인식에 관한 연구”, 전자공학회 논문지, 제40권, 제5호(2003), 266~279.
- [11] Robert M. Haralic, Linda G. Shapiro, “Computer and Robot Vision”, Addison-Wesley, Washington, 1992.
- [12] Paker, “Algorithms for Image Processing and Computer Vision”, John Wiley & Sons, New York, 1997.
- [13] Morton Nadler and Eric P. Smith, “Pattern Recognition Engineering”, John Wiley & Sons Inc. 1993.
- [14] M. Bokser, “Omnidocument technologies”, Proceedings of the IEEE, Vol.80, No.7(1992), 1066~1078.
- [15] D. Trier and A. K. Jain and T. Taxt, “Feature extraction methods for character recognition a survey”, Pattern Recognition, Vol.29, No.4(1996), 641~662.
- [16] H. Freeman and J. M. Glass, “Boundary Encoding and Processing”, Picture Processing and Psychopictorics, B. S. Lipkin and A. Rosenfeld, eds., Academic Press, New York, 241~263, 1970.

Abstract

Machine-printed Numeral Recognition using Weighted Template Matching with Chain Code Trimming

Jung, Minchul*

This paper proposes a new method of weighted template matching for machine-printed numeral recognition. The proposed weighted template matching, which emphasizes the feature of a pattern using adaptive Hamming distance on local feature areas, improves the recognition rate while template matching processes an input image as one global feature. Template matching is vulnerable to random noises that generate ragged outlines of a pattern when it is binarized. This paper offers a method of chain code trimming in order to remove ragged outlines. The method corrects specific chain codes within the chain codes of the inner and the outer contour of a pattern. The experiment compares confusion matrices of both the template matching and the proposed weighted template matching with chain code trimming. The result shows that the proposed method improves fairly the recognition rate of the machine-printed numerals.

Key Words : Template Matching, Numeral Recognition, Chain Code, Hamming Distance

* Department of Computer Engineering, School of Engineering, Sangmyung University