

# 문자인식을 위한 효율적인 획 정규화

<sup>0</sup>김도현\* 강동구\* 강민정\*\* 차의영\*

부산대학교 전자계산학과

부산대학교 멀티미디어학과

e-mail:{uliminer, dkkang1, dragon, eycha}@harmony.cs.pusan.ac.kr

## An Effective Stroke Normalization for Character Recognition

<sup>0</sup>Do-Hyeon Kim\* Dong-Koo Kang\* Min-Kyeong Kang\*\* Eui-Young Cha\*

\*Dept. of Computer Science, Pusan National University

\*\*Dept. of Multimedia, Pusan National University

### 요 약

문자 인식에 있어서 입력 벡터 생성을 위한 정규화(Normalization)는 다차원 입력 벡터를 단순화하여 속도 및 인식률의 개선을 가져오는 중요한 과정이다. 본 논문에서는 문자 인식을 위한 효율적인 입력 영상의 정규화 방법에 대한 연구와 제안하는 획 정규화 기법을 이용하여 Kohonene Neural Network으로 다양한 입력 영상 변형에 따른 인식 성능을 실험으로 비교 분석하여 제안한 정규화 방법의 효율성을 증명하였다.

### 1. 서 론

현대 사회가 정보화 사회로 진입하고 급변하는 전자 계산기의 발달에 따라 우리의 정보 처리 능력은 급진전하고 있으나, 아직도 대부분의 컴퓨터 자료 입력은 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 많은 연구가 진행되었고[1-4], 그 결과도 매우 우수하게 증명되고 있으나 실질적으로 산업화하여 사용될 경우, 인식기 자체가 그 산업 현장 환경에 특성화되어 있는 경우가 많으므로 범용적으로 일반화되어 사용하기가 힘든 실정이다. 또한 문자 인식기로 많이 사용되고 있는 BP (Backpropagation) Network이나 ART II, Kohonen Self-Organizing Feature Map 과 같은 인공 신경망을 사용할 경우, 신경망의 입력값으로 사용되는 벡터의 정규화 과정에 대한 언급이 부족한 실정이며 신경망 자체의 구조적인 복잡도와 알고리즘의 개선에 치우치는 점을 간과할 수 없다. 즉, 다차원 공간의 입력값들을 효율적으로 정규화하지 못한 입력 벡터가 사용될 경우 인식기의 성능 저하에 지대한 영향을 미치게 된다. 따라서 본 논문에서는 그림 1에서 보는 바와 같이 문자 인식을 위한 선행 단계로서 입력 영상을 효율

적으로 정규화하는 영상처리 방법에 대한 연구로 몇 가지 방법을 제시하고 이를 다양한 종류의 입력 변형을 가지는 숫자 데이터에 대한 실험을 통하여 그 유효성을 증명한다.

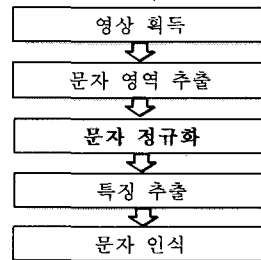


그림 1 문자인식 과정

2장에서는 문자인식을 위한 선행단계로서의 문자 추출방법과 그에 따른 정규화의 문제점을 살펴보고, 3장에서는 추출된 문자 영역에 대하여 제안하는 획 정규화 방법에 대해 설명한다. 4장에서는 Kohonen Network을 사용하여 제안하는 방법에 의해 정규화된 문자에 대한 인식률을 실험을 통해 살펴보고 5장에서는 향후 연구 방향 및 결론을 맺는다.

### 2. 문자 영역 추출 및 영역 정규화 방법

CCD 카메라나 스캐너 등으로 입력받은 문서 이미

지상에서 문자 영역을 추출하기 위하여 흔히 투영(Projection)을 사용한다. 즉, 이진화된 문서 이미지에서 수평방향 투영(Horizontal Projection)을 통하여 문자열을 분리하고 분리된 문자열 부분에 대하여 수직방향 투영(Vertical Projection)을 통하여 각각의 문자를 분리 추출한다. 하지만 이와 유사한 여러 가지 방법을 통하여 그림 2에서 보는 바와 같이 분리된 문자들은 실제적으로 추출된 문자 영역의 위치가 일정하지 않을 뿐만 아니라 너비 및 높이 또한 각각 상이하다.



(a) 숫자 원영상



(b) 투영의 의한 영역 추출



(c) 외접사각형에 의한 영역 추출

그림 2 추출 문자 영역

문자 인식을 위한 입력 영상의 크기 정규화는 위와 같이 상이한 위치와 크기를 갖는 다양한 입력 문자 영상 영역에 대하여 지정된 크기로의 확대/축소 Scaling을 기본으로 하는데 이는 식 (1)에서 볼 수 있는 Scaling Factor<sup>1)</sup>  $S_x$ ,  $S_y$ 에 의한 영상의 기하학적 변환을 의미한다.

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (1)$$

일반적으로 투영에 의해 추출된 영역(그림 2. (b))은 불필요한 공간이 존재하므로 이를 문자 영역에 외접한 사각영역으로 축소한 영역(그림 2. (c))을 Scaling 하거나 외접사각영역에서 통계적 지식에 의한 너비/높이의 비율로 영역을 조정하여 Scaling 한다. 하지만 비율에 의한 Scaling 또한 투영에 의한 방법과 마찬가지로 문자의 너비/높이의 비율이 일정하지 않을 경우 조정된 영역의 상하 또는 좌우로 빈 공간이 발생하게 되므로 비교적 잘 사용되지 않는 방법이다.

### 3. 획 정규화 (Stroke Normalization)

입력 영상의 확대/축소에 의한 크기 정규화는 입력 차원의 수를 낮춤으로써 인식기의 학습 및 인식을 위한 속도 개선의 효과를 가져오는 반면, 중요한 입력 정보를 유실할 가능성이 있다. 따라서 본 논문

에서는 입력 벡터의 생성을 위한 문자 영상의 개선으로 문자 획에 대한 정규화 방법을 다음과 같이 제시한다.

#### 3.1 외접 사각영역 설정

2장에서 살펴본 바와 같은 방법으로 투영 등의 방법에 의하여 대략적인 문자 외곽영역을 구한 후 영역의 안쪽방향으로 획을 만날 때까지 영역을 축소해 나가면 외접사각 영역을 구할 수 있다(그림 3). 이때, 다음 단계에서 수행될 모폴로지 확장 연산을 수행할 경우 외접 부분의 획 화소 유실에 대비하여 외곽영역을 1 픽셀 확대하여 설정한다.

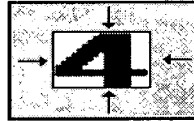


그림 3 외접사각영역

#### 3.2 모폴로지(Morphology)에 의한 획 굵기 조정

문자의 종류(글꼴), 크기, 방향 등에 따라 문자의 획이 차지하는 굵기가 다양하며 이것은 동일한 문자에 대하여 다양한 형태의 입력 변형을 의미한다. 이는 곧 문자 인식기의 강인성(Robustness)을 약화시키며 나아가 오인식의 원인이 되기도 한다. 따라서, 다양한 문자 환경에서도 일관성 있도록 획의 굵기를 조정하기 위해서 확장연산(Dilation)과 침식연산(Erosion)을 혼합한 모폴로지 연산[5]을 조합한 방법을 사용한다. 즉, 외접사각영역의 이진 문자 영상  $f$ 와,  $f$ 가 확장 연산과 침식 연산에 의해 변환된 문자 영상  $f_d$ 와  $f_e$ 의 평균광강도  $M_f = \text{mean}(f)$ ,  $M_{f_d} = \text{mean}(f_d)$ ,  $M_{f_e} = \text{mean}(f_e)$ 를 각각 구한다. 각  $M_f$ ,  $M_{f_d}$ ,  $M_{f_e}$ 에 대해 표준 문자 영상의 평균광강도  $M_s (=128)$ 와의 차이를 계산하여 그 차이가 가장 작은 연산  $\text{MIN}(|M_s - M_f|, |M_s - M_{f_d}|, |M_s - M_{f_e}|)$ 에 의해 수행된 변환 영상  $f'$ ( $f$  or  $f_d$  or  $f_e$ )를 1차 정규화된 영상으로 택한다. 따라서 이와 같은 정규화 과정을 거치면 다양한 굵기를 가지는 문자의 획의 두께가 Morphology Mask에 의한 확장 또는 침식 연산으로 평균적으로 근사화되며, 이것은 문자의 다양한 입력 변형을 흡수하는 효과로 작용하게 된다. 여기서 Morphology Mask는 중심 화소를 기준으로 확장 또는 수축을 결정할 이웃 픽셀들의 위치를 '1'로 설정한 참조배열로 일반적인 3x3 mask를 사용하였다.

#### 3.3 다차원 정규화 반복 및 최종 크기 Scaling

3.2의 과정을 통해 생성된 정규화 이미지( $f'$ )는 입력 문자 영상( $f$ )에 대한 1차 정규화 이미지로  $f$ ,  $f_d$ ,  $f_e$  중 하나의 이미지로 만들어지게 된다.

이때,  $f' = f_d$  이거나  $f' = f_e$  인 경우는 입력 문자 영상에 대한 획 굵기가 상하좌우 각각 1 픽셀 증가되거나 감소되어 영상이 개선된 경우로써 반복적인

1) 정규화 영상의 크기가 20x10이고, 추출된 문자 영역의 크기가 50x20일 때, Scaling Factor  $S_x$ ,  $S_y$ 는 각각 0.5, 0.4이다.

획 굵기 조정으로 좀더 영상을 개선할 여지가 있다. 따라서 이런 영상에 대해서는 추가의 k회 만큼 3.1, 3.2의 과정을 반복 수행하여 새로운 k차 정규화 영상을 생성하되, 모폴로지 확장연산에 의한 획 접합과 침식연산에 의한 획 유실을 방지하기 위해 k 값의 한계를 설정한다(ex. k<3 : 2회 반복 허용)

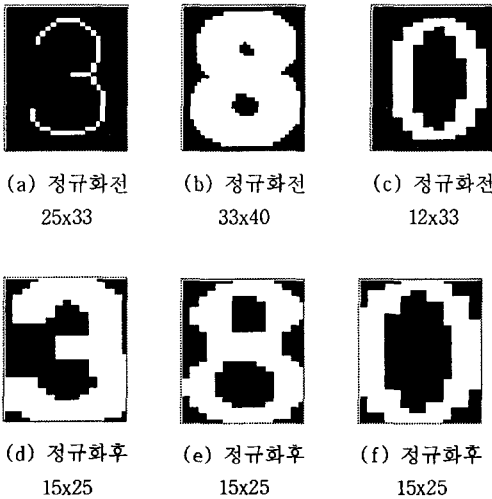


그림 4 다양한 크기의 문자 획 정규화

한편, 더 이상의 반복이 필요없는 f'=f. 인 경우 또는 정규화 반복이 끝난 경우에는 median filtering으로 획의 꺾임 및 계단 현상을 완만하게 하고, 최종적으로 계산되어진 Scaling Factor S<sub>x</sub>, S<sub>y</sub>을 사용하여 원하는 크기로 Scaling을 수행함으로써 그림 4에서 보는 바와 같이 일반화되고 안정적이며 효과적인 정규화가 이루어지게 된다.

4. 실험 및 결과

문자 획 정규화의 유효성을 검증하기 위하여 본 논문에서는 다양한 모양 및 크기를 갖는 숫자 데이터에 대하여 자율신경망인 Kohonen Neural Network을 사용하여 인식을 개선 여부를 실험하였다.

4.1 Kohonen Neural Network

Kohonen Network은 BP (Backpropagation Network)와는 달리 일반적으로 계층적인(hierarchical) 시스템이 아니며 그림 5에서 보는 바와 같이 2개의 층으로 이루어져 있는데 첫 번째 층은 입력층(input layer)이고 두 번째 층은 경쟁층(competitive layer)이다. 입력층과 경쟁층 사이의 연결선은 연결강도를 나타내며 양층 사이에 완전연결(fully connected)되어 있는 구조이다. 이와 같은 구조의 Kohonen Network에서의 학습은 다음과 같은 단계로

이루어진다[6].

- [단계 1] 연결강도를 초기화한다.
- [단계 2] 새로운 입력벡터를 제시한다.
- [단계 3] 입력벡터와 모든 뉴런들간의 거리를 계산한다. 입력과 출력 뉴런 j사이의 거리 d<sub>j</sub>는 식 (2)와 같이 계산한다.

$$d_j = \sum_{i=0}^{N-1} (X_i(t) - W_{ij}(t))^2 \quad (2)$$

여기서 X<sub>i</sub>(t)는 시각 t에서의 i번째 입력벡터이고 W<sub>ij</sub>(t)는 시각 t에서의 i번째 입력벡터와 j번째 출력 뉴런 사이의 연결강도이다.

- [단계 4] 최소 거리에 있는 출력 뉴런을 선택한다. 최소 거리 d<sub>j</sub>인 출력 뉴런 j\*를 선택한다.
- [단계 5] 뉴런 j\*와 그 이웃들의 연결강도를 재조정한다. 뉴런 j\*와 그 이웃 반경내의 뉴런들의 연결강도를 다음식에 의해 재조정한다.

$$W_{ij}(t+1) = W_{ij}(t) + \alpha (X_i(t) - W_{ij}(t)) \quad (3)$$

여기서 j는 j\*의 이웃 반경내의 뉴런이고 i는 0에서 N-1까지의 정수값이다. α는 0과 1사이의 값을 가지는 이득항(gain term)인데 시간이 경과함에 따라 점차 작아진다.

- [단계 6] 단계 2로 가서 반복한다.

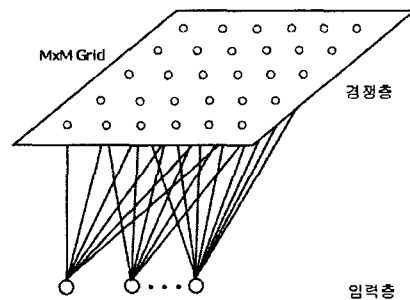


그림 5 Kohonen Network 구조

실험에서 사용된 Kohonen Network의 뉴런 경쟁층은 30x30의 Grid로 구성하고, 입력층의 입력벡터의 수는 입력 영상의 크기를 20x30으로 정규화하여 600개를 사용하였다. 또한 초기 학습률은 0.3으로 하고, 초기 이웃반경은 뉴런 경쟁층폭의 1/2 즉, 30/2 = 15로 정하여 반복 회수에 따라 점차 감소하도록 하였으며, 입력 학습 파일 (220개) 전체에 대한 수행을 1회로 하였을 때, 500회의 반복을 수행하였다.

4.2 실험 데이터

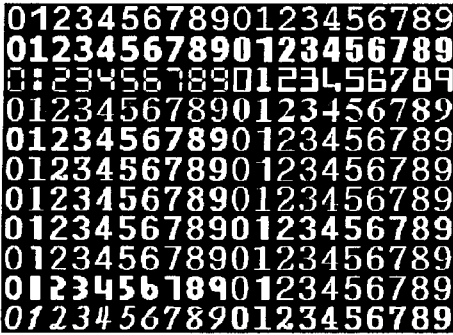


그림 6 학습 숫자 데이터 22종 220개

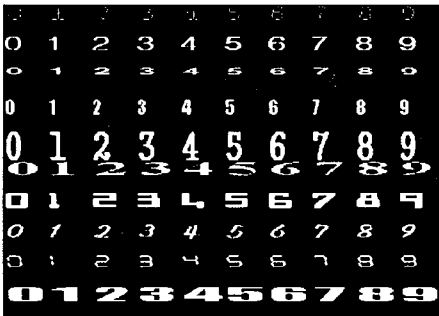


그림 7 인식 숫자 데이터 예

본 논문에서 숫자 인식기로 사용한 Kohonen Network의 학습을 위해 사용한 실험 데이터는 22가지 글꼴으로 총 220개의 숫자 영상 파일(그림 6)을 사용하였다. 그리고 입력 데이터의 다양한 입력 변형을 포함할 수 있도록 위치 이동, 회전(좌우), 세선화, 확장, 확대(상하, 좌우), 축소(상하, 좌우)하여 총 2200개의 숫자 영상 파일 (그림 7, 22종 \* 10(9가지 변형+원형) \* 10 = 2200개)에 대한 인식 실험을 하였다. 그 결과 표 1에서 보는 바와 같이, 외접사각영역에 의한 방법으로 정규화를 한 경우 인식이 93.0% 인데 비해, 제안한 방법의 경우 97.3%로 인식을 향상을 가져오는 것을 알 수 있다.

5. 향후 연구 방향 및 결론

본 논문에서는 문자 인식에 있어서 입력 벡터 생성을 위한 문자 영상의 정규화 방법을 제시하고 실험을 통해 다양한 입력 영상의 변형에 대해서 높은 인식을 보임으로써 제안한 정규화 방법의 효율성을 증명하였다. 본 논문에서 제시한 방법을 좀 더 일반화하기 위하여 획 접합이나 획 유실 현상을 보다 유연하게 처리하는 방법에 대한 연구와 함께 보간법을 사용한 Scaling의 개선 등의 연구가 필요하

2) 사용된 글꼴 22종 : 굴림, 궁서, 돋움, 명조, 바탕, 산세리프, 샘물, 시스템, 신명조, 안상수, 중고딕, 필기, 휴먼체, Arial, Courier, Fixdays, Impact, LED, Orbit-B, Symbol, Times New Roman, Verdana

표 1 실험 결과 비교

숫자 별	외접사각영역 정규화			획 정규화		
	인식률 (%)	정인식 (개)	오인식 (개)	인식률 (%)	정인식 (개)	오인식 (개)
0	99.091	218	2	99.091	218	2
1	96.818	213	7	98.182	216	4
2	91.818	202	18	98.182	216	4
3	97.727	215	5	96.364	212	8
4	92.727	204	16	95.455	210	10
5	99.091	218	2	99.091	218	2
6	89.545	197	23	95.909	211	9
7	85.000	187	33	95.909	211	9
8	86.364	190	30	96.364	212	8
9	92.727	204	16	98.636	217	3
전 체	93.091	2048	152	97.318	2141	59

며 추후 영문과 한글 등의 문자에 대한 검증도 수행되어야 할 것이다.

참고 문헌

[1] 박창순, 김두영, "오프라인 필기체 숫자 인식을 위한 다양한 특징들의 비교 및 인식을 개선 방안", 한국정보처리학회 논문지, 제3권 4호, 1996년 7월  
 [2] 이성환, 박희선, "고리 투영을 이용한 위치, 크기 및 회전 변형에 부관한 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", 한국정보과학회 논문지, VOL.20, No.3, 1993년 3월  
 [4] B.Y. Choi, S.Y. Bang, "Constraints on handwritten Korean characters to improve the machine readability", International Journal of Industrial Ergonomics 26 (2000) 549-559  
 [5] Ramesh Jain, Rangachar Kasturi, Brian G. Schunck, Machine Vision, McGRAW-HILL INTERNATIONAL EDITION  
 [6] 김대수 著, 신경망 이론과 응용(I) Neural Networks Theory and Applications, 하이테크 정보, pp.170~189  
 [7] Mu-Chun Su, Hsiao-Te Change, "Fast Self-Organizing Feature Map Algorithm", IEEE Transaction on Neural Networks, VOL.11, No.3, MAY 2000.  
 [8] Dong-Chul Park, "Centroid Neural Network for Unsupervised Competitive Learning", IEEE Transaction on Neural Networks, VOL.11, No.2, MARCH 2000