산학기술성공학회논문지
Vol. 3, No. 4, pp. 245-250, 2002

연역적이고 국부적인 영문자의 폰트 분류법

정 민 철*

A Priori and the Local Font Classification

Min Chul Jung*

요 약 본 연구에서는 영문 단어로부터 폰트를 분류하기 위해 연역적이고 국부적인 폰트 분류 방법을 제안한다. 이는 문자 인식 전에 한 단어의 폰트를 분류하는 것을 말한다. 폰트 분류를 위해 활자 특성인 Ascender, Descender와 Serif가 사용된다. 입력 단어로부터 Ascender, Descender와 Serif가 추출되어 경사도 특징 벡터가 추출되고, 그 특징 벡터는 인공 신경망에 의해 입력 단어에 대한 폰트 스타일, 폰트 그룹, 폰트 이름이 분류된다. 제안된 연역적이고 국부적인 폰트 분류 방법은 폰트 정보가 문자 분할기와 문자 인식기에 사용될 수 있게 한다. 나아가, 특정 폰트에 따른 Mono-Font 문자 분할기와 Mono-Font 문자 인식기로 구성되는 OCR 시스템을 구성할 수 있는 것을 가능하게 한다.

Abstract This paper presents *a priori* and the local font classification method. The font classification uses ascenders, descenders, and serifs extracted from a word image. The gradient features of those sub-images are extracted, and used as an input to a neural network classifier to produce font classification results. The font classification determines 2-font styles (upright or slant), 3-font groups (serif, sans serif, or typewriter), and 7-font names (PostScript fonts such as Avant Garde, Helvetica, Bookman, New Century Schoolbook, Palatino, Times, or Courier). The proposed a priori and local font classification method allows an OCR system consisting of various font-specific character segmentation tools and various mono-font character recognizers.

Key Words : Font Classification, OCR, Artificial Neural Network

1. 서 론

영문으로 인쇄된 문서의 문자인식 분야에 있어서, OCR (Optical Character Recognition) 시스템에 알려진 폰트로 쓰여진 영문자를 인식하는 인식률은 높게 나타나다. 그 이유는 OCR 시스템이 알려진 폰트 내에서 공통된 규칙성을 발견하여 학습하기 때문이다. OCR 시스템에 알려지지 않은 폰트로 쓰여진 문자인식은 폰트의 다양성 때문에 인식률이 전자에 비해 급속히 떨어진다. 다양한 폰트로 쓰여진 문서의 문자인식에서 높은 인식률을 계속 유지하는 OCR 시스템을 만드는 것은 아직 풀어야 할 연구 과제이다. 이 문제를 해결하기 위해 Omni-Font OCR 시스템이 소개되었는데, 이 시스템은 한 문자에 대해 폰트의 분류 없이 일반적인 특징 벡터를 추출하여 문자 인식을 한다 [1]. 그러나 이 논문에서 제시하는 폰트의 분류는 분류된 특정 폰트 내에 속한 문자 구조와 활자 특성에 대한 정보를 얻을 수 있다. 이러한 폰트 정보는 분류된 특정 폰트에 따른 문자인식을 하는 OCR 시스템을 구성할 수 있게 한다. 더 나아가 분류된 특정 폰트에 따른 문자 분할을 가능하게 한다. 즉, 폰트를 분류하면 특정 폰트에 따른 Mono-Font 문자 분할기와 Mono-Font 문자 인식기로 구성되는 OCR 시스템을 구성할 수 있다. Figure 1은 이 논문에서 제안된 폰트 분류기의 역할과 여러 개의 Mono-Font 문자분할 모듈기와 Mono-Font 문자 인식기로 구성되는 OCR 시스템을 나타낸다.

*상명대학교 컴퓨터시스템공학과

2. 폰트 분류의 접근법

이 연구에서 폰트분류를 위해 사용한 접근법은 연역적이고 국부적인 접근 방식이다(a priori and the local approach). 폰트 분류기에 위치에 따라 폰트 분류는 연역적인 방법과 귀납적인 방법으로 구분된다. 이 연구에서 제안된 연역적인 접근법은 문자 인식 전에 문자의 폰트를 알아내는 접근법이다. 따라서 폰트 분류기는 문자 인식전 전처리 단계(Preprocessing Step)에 위치하여,

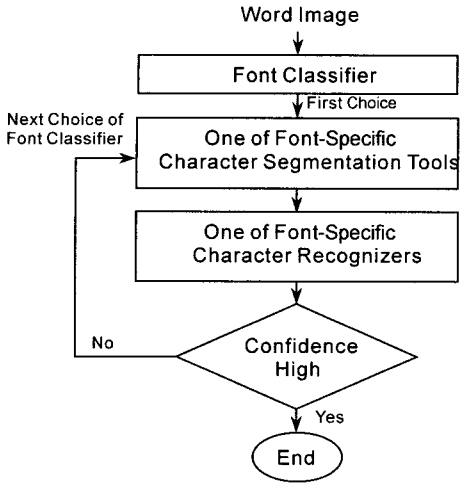


Figure 1. 연역적이고 국부적인 폰트 분류(a priori and the local font classification).

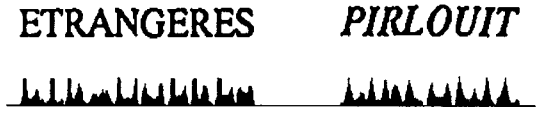
폰트 정보가 문자 분할기와 문자 인식기에 사용될 수 있다. 귀납적인 접근법(a posteriori approach)은 문자 인식 후에 문자 인식기의 결과를 이용하여 폰트를 분류한다[2, 3]. 폰트 정보는 문자인식 후, 문서를 원래의 폰트로 표현할 때 사용되어진다. 또한 폰트 분류기의 입력 텍스트의 길이에 따라 폰트 분류는 국부적인 방법과 전체적인 방법(the global approach)으로 구분된다. 이 연구에서 제안된 국부적인 접근법은 폰트 분류기의 입력으로 한 단어(a word)나 한 문자(a character)를 사용한다. 물론 스캔된 상태의 질이 떨어지는 문서에서는 한 단어나 한 문자의 입력으로 정확한 폰트를 분류하는 것은 심지어 인간에게도 불가능하다. 그러나 이 연구에서 제안된 폰트 분류법은 그러한 단어와 문자로부터도 최소 Serif 폰트 그룹과 Sans Serif 폰트 그룹으로 분류하는 것이 가능하다. 전체적인 접근법(the global approach)은 한 페이지나 한 구문의 문서 전반에 걸쳐 주로 사용된 폰트를 분류하는 접근 방법이다. S. Khoubyari는 한 문서에서 쓰여진 폰트를 분류하기 위해 영문 문서에서 사용빈도수가 높은 단어들인 'the', 'of', 'and', 'a', 'to' 등을 사용하여 문서 전체에 사용되어진 폰트를 분류하였다[4]. 귀납적 폰트 분류와 전체적인 폰트 분류는 OCR 시스템의 결과를 이용한 방법으로 OCR 시스템의 인식률을 높이는 데 이용하기가 용이하지 않다. 그러나 본 연구에서 제안된 연역적이고 국부적인 폰트 분류는 OCR 시스템의 전처리 단계에서 폰트를 분류하여, 분류된 폰트 정보를 OCR 시스템이 문자 분할과 문자인식을 수행할 때 이용 가능하게 하여 OCR 시스템의 인식률을 높이는 데 이용될 수 있다. 나아가 Figure 1에서와 같이 특정 폰트의 문자에 강한 문자 분할기와 문자 인식기를 병렬로 구성하는 OCR 시스템을 구성할 수 있다. 연역적이고 국부적인 폰트 분류법을 사용하는 본 연구에서는 한 단어를 입력으로 받아, 2 폰트 스타일(Upright와 Slant), 3 폰트 그룹(Serif, Sans Serif와 Typewriter), 7 포스트스크립트 폰트(Avant Garde, Helvetica, Bookman, New Century School Book, Palatino, Times와 Courier)를 문자 분할과 분자 인식 전에 분류한다. 위의 포스트스크립트 폰트는 레이저 프린터에 널리 표준으로 쓰이는 폰트이다. 폰트 분류기가 사용되는 응용 분야에 따라 위의 폰트는 삭제 또는 첨가해 나갈 수 있다. 또한 폰트 분류기에 알려지지 않은 새로운 폰트는 위의 폰트에 가장 가까운 폰트로 분류된다. 이 연구에서는 폰트 분류의 실험을 위해 Gradient 특징 추출법과 인공 신경망이 이용되었다.

3. 폰트 분류의 방법

3.1 수직 투영 윤곽 분석(Vertical Projection Profile)에 의한 폰트 스타일 분류

OCR 시스템에 입력된 문자가 Upright(혹은 Normal) 스타일인지 Slant(혹은 Italic) 스타일인지 분류하는 것은 중요하다. 왜냐하면, 먼저 문자를 분할할 때, 폰트 스타일에 따라 문자 분할 방식이 달라야한다. 예를 들면 Upright 스타일로 쓰여진 접합 문자는 수직으로 문자 분할이 가능하나, Slant 스타일로 쓰여진 접합 문자는 수직 문자 분할이 불가능하다. 또한 문자를 인식할 때, 문자의 내부 구조가 폰트 스타일에 따라 다르다는 것을 주시해야한다. 예를 들면 Upright로 쓰여진 'm'과 Slant로 쓰여진 'm'은 문자의 구조(또는 모양)가 다르다. 폰트 스타일을 분류하기 위해 Figure 2와 같이 수직 투영 윤곽 분석법(the Vertical Projection Profile)을 사용하였다.

Figure 2에서 볼 수 있는 것처럼 Upright 스타일과 Slant 스타일의 수직 투영 윤곽은 완전히 다른 모양을 하고 있다. Upright 스타일로 쓰여진 단어의 수직 투영 윤곽에서는 직사각형의 봉우리를 발견할 수 있고, Slant 스타일로 쓰여진 단어의 수직 투영 윤곽에서는 삼각형의 봉우리를 주로 발견할 수 있다. 이 두 스타일의 수직



(a) Upright (b) Slant

Figure 2. Upright와 Slant 폰트 스타일의 수직 투영 윤곽 분석.

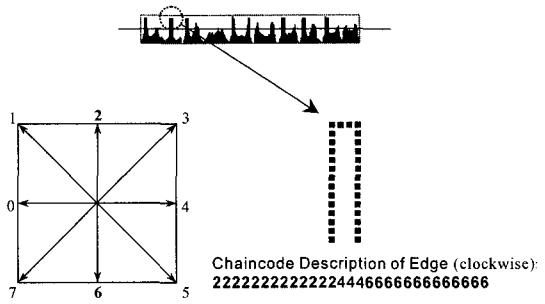


Figure 3. 직사각형 봉우리의 체인코드 분석.

투영 윤곽을 분류하기 위해 Figure 3에서처럼 체인코드가 사용되었다.

Figure 3은 Upright 스타일의 수직 투영 윤곽 이미지를 상단 부와 하단 부로 나눈 후 상단 부에서 얻을 수 있는 직사각형의 체인 코드를 나타낸다. 직사각형의 체인코드는 연속적인 수직 방향 코드로 구성되는 반면, 삼각형의 체인 코드는 연속적인 수직 방향 코드가 구성되지 않는다.

3.2 Serif에 의한 폰트 그룹 분류

영문자에 있어 Serif는 문자의 주 획의 위아래 양끝에 붙어 문자를 꾸미는 작은 획이다. Sans Serif는 Serif가 없다는 뜻이다. Serif는 폰트를 분류하는 데 가장 큰 특징 중 하나이다. 예를 들면 Serif 폰트 그룹에 속하는 Times 폰트의 'I', Sans Serif 폰트 그룹에 속하는 Helvetica 폰트의 'I', Typewriter 폰트 그룹에 속하는 Courier 폰트의 'I'는 그 모양에 있어 크게 다르다. Figure 4는 포스트스크립트 폰트의 문자 하단 부에서 추출한 다양한 Serif 모양을 나타낸다.

Serif의 존재 여부에 따라 폰트는 Serif 폰트 그룹과 Sans Serif 폰트 그룹으로 나눌 수 있는데 Serif 폰트 그룹에는 Bookman, New Century School Book, Palatino와 Times 폰트가 있고, Sans Serif 폰트 그룹에는 Avant Garde와 Helvetica 폰트가 있다. Courier 폰트의 문자도 Serif를 가지나 Figure 4에서 볼 수 있듯이 Serif가 주 획의 굵기와 같으며 Serif 크기 또한 다른 폰트와 비교할 때 제일 크다. Courier 폰트는 원래 타자기의 폰트로서 각 문자의 실제 폭에 관계없이 Serif를 이용하여 모든 문자의 폭을 같게 만든 것이 특징이다. 예를 들면 Courier 폰트로 쓰여진 'I'와 'W'는 여백을 점유하고 있는 폭이 동일하다. 즉, 같은 폭을 점유하기 위해 실제 폭이 적은 'I'의 Serif를 옆으로 길게 강조했다. 이와 같이 각 문자의 실제 폭과 관계없이 일정한 문자 폭을 가지는 Typewriter 폰트는 Fixed-Pitch라고 하는데 접합문자가 드물며 문자 접합 시에도 문자 분할은 일정한 비에 따라 기계적으로 수행 가능하다. 그러나 컴퓨터의 등장에 따른 Variable-Pitch로 쓰여진 접합문자의 경우 문자 분할은 문자 구조의 선지식을 요구한다. 따라서 본 연구에서는 폰트를 3가지 그룹, Serif 폰트 그룹, Sans Serif 폰트 그룹과 Typewriter 폰트 그룹으로 분류하였다.

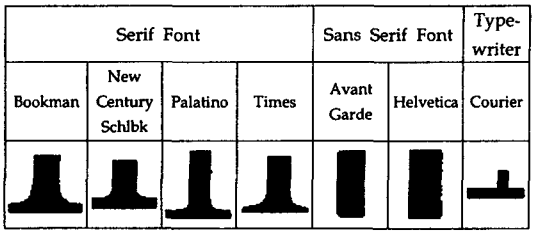


Figure 4. 포스트스크립트 폰트의 문자 하단 부에서 추출한 다양한 Serif.

3.3 Serif, Ascender와 Descender를 이용한 폰트 분류

영문 알파벳의 문자 구조적 특징 중 하나는 영문 문자들이 Ascender나 Descender를 가진다는 것이다. Figure 5(b)에서 볼 수 있는 것과 같이, 단어의 수평 투영 윤곽 분석을 하면 3가지 영역, 즉 Ascender 영역, x-height 영역과 Descender 영역으로 나눌 수 있다. 이 세 가지 영역은 Ascender 라인, x-height 라인, Base 라인과 Descender 라인으로 구분된다. x-height 영역은 모든 영문 단어가 점유하나 Ascender 영역과 Descender 영역은 영문 단어에 따라 점유 될 수도 있고 안될 수도 있다. 이러한 영역의 분석 방법을 통해 각 문자의 Serif, Ascender와 Descender의 활자 특성을 추출한다. 추출하는 방법은 소문자로 구성된 단어와 대문자로 구성된 단어, 2가지로 구분된다.

3.3.1 소문자의 폰트 분류

소문자로 구성된 단어는 'Ascender 영역과 x-height 영역' 또는 'x-height 영역과 Descender 영역'의 두 가지 영역만을 가지거나 세 가지 영역 모두를 가질 수 있다. Figure 5는 세 가지 영역 모두를 가지는 단어의 예를 보였다. Figure 5(a)에서처럼 처음 문자가 대문자인 단어(Capitalized Word)도 이 구분에 속한다.

Figure 5(c)는 x-height 영역을 제거한 후 남은 이미지이다. 이 이미지로부터 Serif인 ①과 Descender인 ②를 얻을 수 있다. 'T' 상단부분은 Ascender의 후보로 검증되나 그 폭이 Ascender가 되기에는 너무 커서 제외된다. Serif를 추출하기 위해, Figure 5(d)에서와 같이

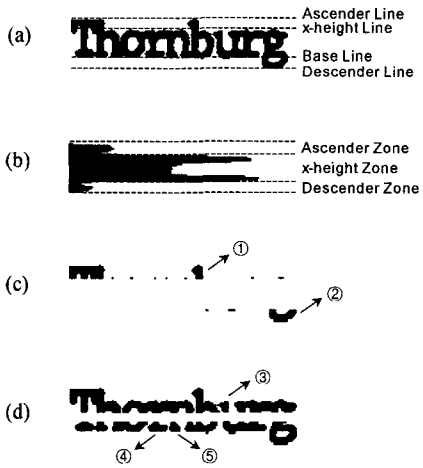


Figure 5. 소문자로 구성된 단어의 활자 특성 추출: (a) Capitalized Word, (b) 입력 단어의 수평 투영 윤곽 분석, (c) x-height 영역을 제거한 후 남은 이미지, (d) x-height의 중앙부분을 제거하고 남은 이미지.

x-height의 중앙부분을 제거하고 남은 이미지를 분석한다. 그 이미지로부터 Serif인 ③, ④와 ⑤를 얻을 수 있다. 나머지들은 Serif가 되기에는 폭이나 높이가 문자 크기와 비교할 때 너무 커서 모두 제외된다. 접합 문자에서 발생하는 접합된 Serif는 이 추출 방법에서 모두 제외된다.

3.3.2 대문자의 폰트 분류

모두 대문자로만 구성된 단어는 Figure 6(b)의 수평 투영 윤곽 분석에서 볼 수 있는 것과 같이 한 영역만을 가진다. 따라서 단어의 중앙부분을 제거하고 남은 이미지를 분석하면 Serif를 얻을 수 있다. ①부터 ⑪은 모두 Serif 후보가 된다. 나머지들은 모두 Serif가 되기에는 문자의 크기와 비교 할 때 모두 너무 폭이 넓어서 제외된다. 또한 본 연구에서 Serif는 대칭성을 가진 것으로 한정하였다. 모양에서 비대칭인 Serif 후보를 제외하면, Figure 6(c)에서 ①, ④, ⑤, ⑦과 ⑩이 Serif로 선택된다.

3.4 경사도 특징 추출

경사도 특징 추출(Gradient Feature Extraction)은 필기체 인식을 위해 개발되었다[5, 6]. 본 연구에서는 추출된 활자 특성인 Ascender, Descender와 Serif들로부터 경사도 특징 추출이 수행된다. 경사도는 Sobel 연산자에 의해 수행된다. 추출된 활자 특성 이미지의 각 픽셀은 Sobel 연산자 템플릿과 Convolution되어 Δx-성분과 Δy-성분 값이 중앙 픽셀로 그 값이 저장되어진다. 즉, 중앙 픽셀의 경사도는 주변의 이웃 8-픽셀의 함수로서 계산되어진다. 경사도는 0에서 2π라디안의 범위를

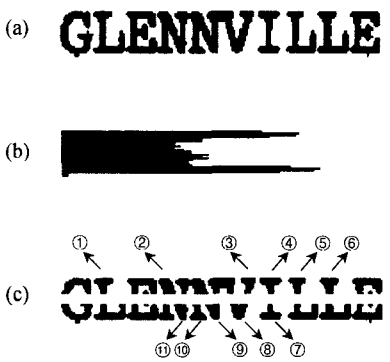
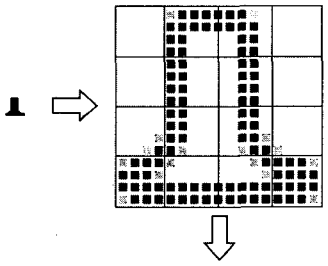


Figure 6. 대문자로만 구성된 단어의 활자 특성 추출: (a) All Capitals, (b) 입력 단어의 수평 투영 윤곽 분석, (c) 입력 단어의 중앙부분을 제거하고 남은 이미지.

가지며 계산의 단순화와 속도를 감안해 8개 영역으로 양자화하고 정반대 방향은 같은 것으로 간주하여 4개의 경사도를 나타내면 경사도는 0, π/4, π/2와 3π/4로 나타낼 수 있다. 추출된 활자 특성 이미지는 4×4 grid로 영역을 나누고, 각 영역에 있는 픽셀의 경사도를 히스토그램으로 누적 수치화 하여 지정된 threshold 값을 초과하는 경사도를 카운트한다. 이러한 방법은 4×4×4, 즉 64개의 특징 벡터를 구성한다. 이 특징 벡터는 인공 신경망의 입력 벡터가 된다. Figure 7은 추출된 serif의 경사도 맵과 그 특징 벡터를 나타낸다.

4. 인공 신경망 Classifier

폰트 분류를 위해 인경 신경망 Classifier가 이용되었



0.00	0.00	0.50	0.00	0.33	0.00	0.00	0.00
0.00	0.00	0.17	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	1.00	0.00	1.00	0.00	0.00	0.00
0.17	0.00	0.00	0.50	0.17	0.00	0.17	0.00
0.17	0.00	0.33	0.00	0.50	0.50	0.00	0.00
0.33	0.00	1.00	0.00	1.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00

Figure 7. 추출된 serif의 경사도 맵과 특징 벡터

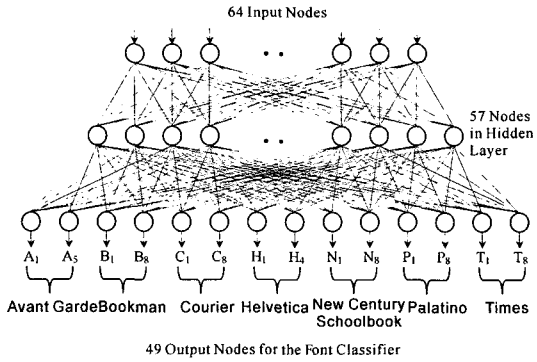


Figure 8. 3-Layer Back propagation 인공 신경망 구성도.

다. 즉 폰트 분류기는 64개의 입력 노드를 가지고, 57개 노드의 hidden layer를 가지며, 49개의 출력노드를 가지는 3-layer back propagation 인공 신경망으로 구성된다. Figure 8는 그 구성을 나타낸다. Hidden layer의 노드 개수는 입력노드와 출력 노드수의 평균값으로 초기에 주어진 후, 트레이닝과 테스팅의 절차를 거쳐 가장 최적화 된 값인 57을 선택하였다. 입력 단어로부터 모든 Ascender, Descender 와 Serif가 처리된 후 폰트 분류기는 그 단어에 대한 폰트를 분류한다. 예를 들어 입력 단어로부터 한 개의 Ascender, 한 개의 Descender 와 세 개의 Serif를 추출했다면 추출된 5개의 활자 특성 이미지는 5개의 특징 벡터 세

트로 바뀌어 인공 신경망에 입력되며, 49개의 출력 노드는 누적된 Confidence값을 가진다. Times 폰트의 경우 누적된 Confidence값은 다음과 같이 계산되어진다.

$$Times = \sum_{i=1}^{8} \left(\sum_{j=1}^{n} C_j(T_i) \right) \quad (1)$$

위 식에서 n은 추출된 활자 특성이미지 개수이며, Cj(Ti)는 Times 폰트의 각 Confidence값이며, 8은 Times 폰트의 전체 활자 특성이미지 개수이다(2 ascenders, 4 descenders와 2 serifs). 각 누적된 Confidence 값 중 최대 값을 가지는 폰트가 입력 단어의 폰트로 분류된다.

5. 실험과 결과

실험에 사용된 이미지는 레이저 프린터로 인쇄되어, 300 dpi로 스캔되고, 같은 파라미터를 가지고 이미지 이진화가 되어 얻어졌다. 트레이닝 데이터는 2120개의 이미지인데 활자 크기는 10, 12와 14 point size가 혼합되었다. 테스팅 데이터는 1060개의 이미지인데 역시 0, 12와 14 point size가 혼합되었다. 각 이미지는 폰트 사이즈에 관계없이 일정한 크기를 유지하게 하기 위해 크기

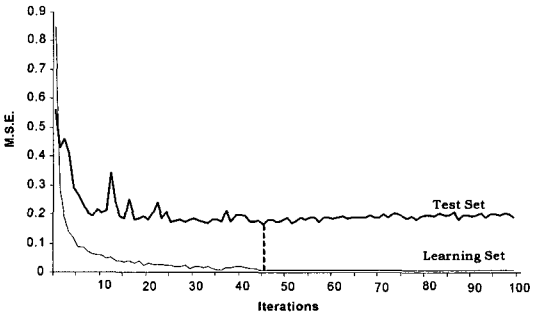


Figure 9. 인공 신경망의 Learning Curve와 Test Curve.

정규화를 하였다. 인공 신경망의 트레이닝은 learning rate = 0.1과 momentum = 0.5로 하여 수행되었다. 트레이닝 과정은 MSE (Mean Squared Error)에 의해 모니터 되었으며, 트레이닝 반복학습(iteration)의 함수로서 MSE의 값을 Figure 9에 나타내었다. MSE 값은 57개의 hidden 노드에서 가장 적은 값을 나타냈으며 0.0069에 수렴하였다. 트레이닝 결과는 테스팅 데이터에 의해 평가되어진다. 테스팅 과정 또한 MSE에 의해 모니터 되어졌다.

테스팅 반복학습(iteration)의 함수로서 MSE의 값을 Figure 9에 나타내었다. 테스팅 과정에서 인공 신경망은 46번의 반복학습에서 MSE의 국부 최소값(Local minima) 0.1648을 가진다. 따라서 46번 반복 학습한 트레이닝의 Weight Matrix가 인공 신경망을 최대로 일반화를 하는 것으로 선택되어진다. 이 값 이상의 트레이닝은 지나치게 학습(over-trained)된 것이다. Table 1은 선택된 폰트에 대한 폰트 분류 결과를 나타낸다. 폰트 분류는 트레이닝 이미지와 테스팅 이미지와 다른 1000개의 단어 이미지로 수행되었다. 그 결과 평균 95.4 퍼센트의 높은 폰트 분류율을 보였다. 10-point size로 쓰여진 단어의 폰트 분류율은 다소 낮은데, 그 이유는 작은 폰트로 쓰여진 문자의 serif 또한 그 모양이 너무 작아 불분명하고 외부 노이즈에 민감하기 때

Table 1. 폰트분류기의 평균 분류율 (단위: 퍼센트)

Point Size		10 pts	12 pts	14pts
Serif Font	Bookman	91.2	96.7	96.8
	N.Century Schlbk	93.5	95.0	97.8
	Palatino	91.1	94.3	97.2
	Times	93.3	95.9	97.3
Sans Serif Font	Avant Garde	94.6	97.4	98.5
	Helvetica	94.1	96.5	97.2
Typewriter	Courier	94.6	95.4	97.0

문이다. Serif 폰트들이 Sans Serif 폰트들보다 다소 낮은 분류율을 보이는 데, 이는 Serif가 더 복잡한 구조를 가지기 때문이다. 폰트 사이즈가 커짐에 따라 폰트 분류율도 증가됨을 볼 수 있다. 폰트 분류의 에러는 주로 같은 폰트 그룹 내에서 발생된다. 그러한 에러는 문자 분할기와 문자 인식기에게 있어 종종 받아들일 수 있는 에러이다. 왜냐하면 같은 폰트 그룹 내에 있는 문자의 경우 그 문자의 구조가 비슷한 경우가 많기 때문이다.

6. 결 론

본 연구에서는 영문 단어로부터 폰트를 분류하기 위해 연역적이고 국부적인 폰트 분류 방법을 제안했다. 폰트 분류를 위해 활자 특성인 Ascender, Descender와 Serif가 사용되었다. 따라서 폰트 분류의 정확도는 Ascender, Descender와 Serif에 의존한다. 적절한 해상도(200dpi 이상)를 가지고 스캔된 문서에 대해서는 실험의 결과에서 나타내듯이 높은 폰트 분류율을 보인다. 이 폰트 분류의 결과는 OCR 시스템의 문자 분할 모듈이나 문자 인식 모듈에서 이용되어 OCR 시스템의 전체 인식률을 높이는 데 이용될 수 있다.

참고문헌

[1] S. Kahan and T. Pavlidis and H.S. Baird, on the recognition of printed characters of any font and size, IEEE transactions on Pattern Analysis and Machine Intelligence, Vol. 9, No. 2, pp.274-288, 1987.

[2] H. Shi and T. Pavlidis, Font Recognition and Contextual Processing for more accurate text recognition, 4th International Conference on Document Analysis and Recognition, pp. 39-41, 1997.

[3] A. Zramdini, Study of Optical Font Recognition based on Global Typographical Features, Ph.D. Dissertation, University of Fribourg, Switzerland, 1995.

[4] S. Khoubyari and J.J. Hull, Font identification using visual global context, SPIE Vol. 2181 Document Recognition, pp. 116-124, 1994.

[5] J.T. Favata, A Multiple Feature/Resolution Approach to Handprinted Digit and Character Recognition, International Journal of Imaging Systems and Technology, Vol. 7, pp. 304-311, 1996.

[6] G. Srikantan, S.W. Lam and S.N. Srihari, Gradient-based Contour Encoding For Character Recognition, Pattern Recognition, Vol. 29, No. 7, pp. 1147-1160, 1996.