

대화식 클러스터링 기법을 이용한 칼라 지도의 문자 영역 추출에 관한 연구

안 창[†] · 박 찬 정^{††} · 이 상 범^{†††}

요 약

지도 영상에 포함된 문자들을 효율적으로 인식하기 위해서는, 문자 영역만을 추출하여 독립된 계층으로 만드는 전처리 단계가 선행되어야 한다.

본 논문에서는 칼라 지도 영상의 색 정보를 대화식 클러스터링 기법에 의해 색 별로 분리하고, 문자 영역을 추출하는 알고리즘을 제안하였다. 제안된 대화식 클러스터링 기법은 칼라 영상 중에서 사용자가 분리하기 원하는 색을 대화식으로 선택하여 각각의 선택된 색을 중심으로 클러스터를 형성하고 분리한다. 문자를 추출하는 알고리즘은 문자를 표현하는 색만을 추출한 이진 영상을 벡터 데이터화 한 후 각 원시 요소들의 상호 관계를 고려하여 직선, 원, 문자의 세 계층으로 분류하는 것으로서, 문자와 직선이 중첩되어, 하나의 연결 요소를 이루고 있을 때에도 효과적으로 문자를 분리할 수 있다.

알고리즘의 평가를 위해 1:3,000의 지번약도에 대한 실험을 수행하였으며, 선분과 문자가 중첩되어 있는 경우에도 각각 분리할 수 있음을 확인하였다.

Character Extraction from Color Map Image Using Interactive Clustering

Chang Ahn[†] · Chan Jung Park^{††} · Sang Burm Rhee^{†††}

ABSTRACT

The conversion of printed maps into computerized databases is an enormous task. Thus the automation of the conversion process is essential. Efficient computer representation of printed maps and line drawings depends on codes assigned to characters, symbols, and vector representation of the graphics.

In many cases, maps are constructed in a number of layers, where each layer is printed in a distinct color, and it represents a subset of the map information. In order to properly represent the character layer from color map images, an interactive clustering and character extraction technique is proposed.

Character is usually separated from graphics by extracting and classifying connected components in the image. But this procedure fails, when characters touch or overlap lines - something that occurs often in land register maps. By vectorizing line segments, the touched characters and numbers are extracted.

The algorithm proposed in this paper is intended to contribute towards the solution of the color image clustering and touched character problem.

※본 논문은 한국과학재단의 연구지원(과제번호: 95-0100-11-01-3)에 의한 연구 결과임.

† 정 회 원: 단국대학교 전자공학과 박사과정

†† 정 회 원: 강릉대학교 전자계산학과 부교수

††† 정 회 원: 단국대학교 컴퓨터공학과 교수

논문접수: 1996년 5월 25일, 심사완료: 1996년 10월 29일

1. 서 론

데이터베이스로 구축된 지리 정보는 교통, 행정, 산업, 군사 등의 분야에 다양한 용도로 활용될 수 있다. 최근 지리 정보 시스템(GIS: Geographic Information System)의 구축이 활발히 이루어지고 있으나, 현재 사용되고 있는 디지털타이저를 사용한 대화식 입력 방법은 많은 시간과 비용이 소요되어, 지리 정보의 수치화에 걸림돌이 되고 있다[1][2]. 이러한 이유로 지도의 자동 입력에 대한 연구가 크게 중요시되고 있다. 자동 입력은 인식 오류라는 심각한 결점을 지니고 있지만, 알고리즘 및 하드웨어의 성능 향상에 따라 인식률이 향상되고 있다.

칼라 영상 처리의 장점은 대체적으로 다음 두 가지로 요약할 수 있다. 첫째로는 영상에서 물체를 식별하거나 분리하기에 적합하다는 점이며, 둘째로는 인간이 물체를 식별하는 방법과 유사하다는 것이다. 실제 인간의 시각 능력은 20가지 정도의 흑백 농도를 식별함에 비해, 색(color)은 수천 가지를 인지할 수 있는 것으로 알려져 있다[3].

지도 영상을 살펴보면, 비교적 단순한 몇 가지의 색 계층으로 이루어져 있음을 알 수 있고, 각각의 색은 적은 수일지라도 각 계층에서 중요한 의미를 가지게 된다. 따라서 이러한 색을 효율적으로 분리할 수 있다면, 보다 효과적으로 지도 영상의 전처리 및 인식 결과에까지 좋은 영향을 미칠 수 있다.

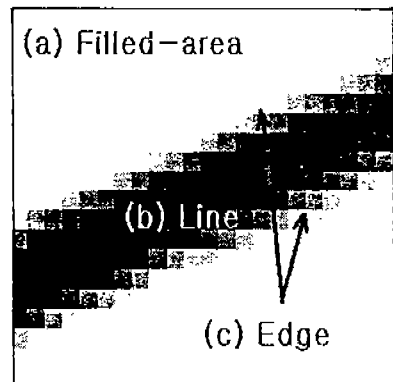
지도의 수치화에 있어서, 지도에 포함된 정보들을 구성 요소별로 묶어 계층별로 분리하는 것은 각 요소의 효율적인 인식과 검색의 편의성을 위해 필수적이다. 이러한 정보의 계층화 과정에 있어 점(point) 단위의 래스터 영상(raster image)을 사용하는 방법은 계산량과 복잡성에 있어서 상당한 부담이 요구되며, 독립적인 연결 요소로 존재하는 문자에 대해서는 분류와 인식이 이루어질 수 있지만 직선과 중첩된 문자의 경우에는 적절한 처리가 매우 어렵다[4].

본 논문에서는 지도의 자동 입력을 위해, 종이에 인쇄된 칼라 지도를 스캔한 래스터(raster) 영상에 대해 동일 색 정보만을 분리(segmentation)하는 대화식 클러스터링(interactive clustering) 기법을 제안하였으며, 이 기법을 이용하여 문자 영역을 표현하는 계층을 분리하고, 이진 영상(binary image)으로 변환하였다.

또한 클러스터링 결과로 얻어진 이진 영상을 벡터화 한 후, 그 결과를 분석하여 문자 영역을 추출하는 알고리즘을 제안한다. 먼저 세션화된 이진 영상의 선을 추적하여 세그먼트와 노드 데이터를 위한 테이블을 작성하고, 각 세그먼트들을 이웃 세그먼트와의 관계를 고려해 서로 연결시키며, 연결된 직선의 길이와 각도를 이용해 직선에 붙어 있는 문자의 분리를 효과적으로 수행한다. 이때 허프 변환(Hough transform)으로 지정된 크기의 원을 추출하여 문자의 추출을 지원한다.

2. 칼라 지도 영상의 전처리

칼라로 인쇄된 지번약도의 특성을 살펴보면, 여러 구성 요소(문자, 선, 배경 등)들이 중첩되어 인쇄되어 있기도 하며, (그림 1(a))와 같이 인쇄 상태에 따라 사람에게서는 단색으로 인식되는 칼라 영역(filled area)이 다양한 색이 혼합된 형태로 표현되기도 하며, 스캐너로 입력될 때 발생하는 다양한 상태의 잡음이 (그림 1(b))와 같은 선 주위에 (그림 1(c))와 같이 경계 부분에 존재하기도 한다.



(그림 1) 확대된 칼라 지도
(Fig. 1) Enlarged of color map

입력된 칼라 영상의 색 정보를 분리하기 위해 일반적으로 많이 사용되는 클러스터링 알고리즘을 적용하는 경우에는 클러스터를 구성하는 화소(pixel)의 개수가 많은 것을 의미하는 정보로 간주하여 분할 과정에서 이용하므로, 클러스터의 중심 자체가 이동하게

되고, 개수가 적은 데이터의 경우에는 오분류의 가능성이 높아지게 된다. 일반 영상과는 달리 특히 지도 영상의 경우에는 클러스터를 구성하는 화소의 수가 적다고 할지라도 중요한 지형 정보가 될 수 있으므로 클러스터링 알고리즘을 수행할 때 고려하여야 한다.

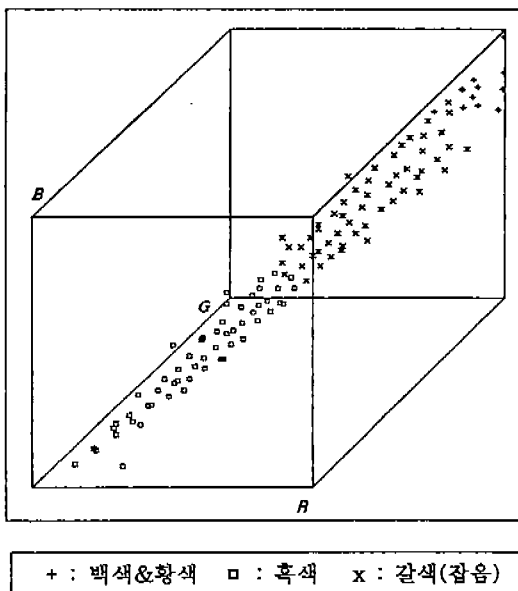
2.1 대화식 클러스터링 기법

현재 시중에서 구입할 수 있는 1:3,000 축척의 지번약도와 같은 칼라 영상의 경우에 도로 영역은 백색, 필지 영역은 옅은 황색, 지면등은 흑색으로 되어 있으며, 행정구역은 적색으로 표시되어 있다. 따라서 도로 영역을 추출하기 위해서는 지번약도 영상에서 백색 부분만을 포함하는 클러스터를 생성하여 분리하면 처리가 용이하게 될 수 있다. 그러나 기존의 K-means 알고리즘과 같은 클러스터링 기법을 적용하면 클러스터의 중심이 클러스터를 구성하는 개체들의 평균값으로 정해지게 되어 소수의 의미있는 정보는 무시되거나 오분류될 수 있고, 원하지 않는 정보가 클러스터로 분리되기도 한다[5].

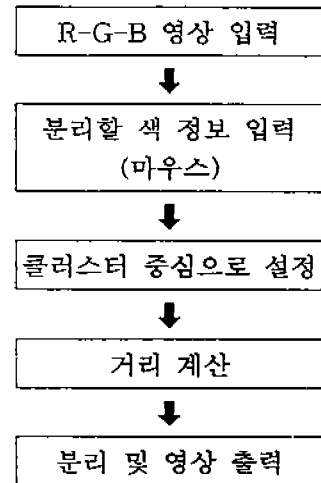
(그림 2)에는 1:3,000 축척의 지번약도 영상을 24비트 칼라 영상으로 입력하여 K-means 알고리즘으로 클

러스터링을 수행한 결과를 보였다. 이때, 클러스터의 개수는 육안으로 보이는 색의 종류인 흑색, 황색, 백색, 3가지로 하였다. 그러나 문자와 선분을 표현하는 흑색의 경우에는 배경을 이루는 백색이나 옅은 황색과 중심 거리가 멀리 떨어져 있으나, 백색과 옅은 황색의 경우에는 서로 인접되어 존재하므로, 분리가 어려워 출력 영상에서도 두 가지 색상이 동일한 클러스터에 소속되었고, 흑색 주위의 잡음 성분인 갈색이 다른 하나의 클러스터를 형성하여 출력됨을 알 수 있다.

제안된 대화식 클러스터링 기법은 (그림 3)과 같다. 먼저 사람이 지도에서 특정 색으로 표현된 정보를 찾는 방법과 유사하게, 사용자가 칼라 지도 영상에서 분리하기 원하는 색상이 표현된 지점을 입력 장치인 마우스(mouse)를 이용하여 각각 지정한다. 이때 지정된 지점의 r, g, b 데이터 값을 각 클러스터의 중심값으로 정하여 두고, 입력 영상의 각 화소의 데이터 값과 각 클러스터의 중심값 사이의 유클리드 거리(Euclidean distance)를 계산하여 가장 근접된 클러스터에 입력 화소를 소속시키면, 3차원 r, g, b 공간 상에서 사용자가 지정한 색상만으로 구성된 클러스터를 형성시킬 수 있다.



(그림 2) R-G-B 좌표계에서의 클러스터링
(Fig. 2) Result of clustering

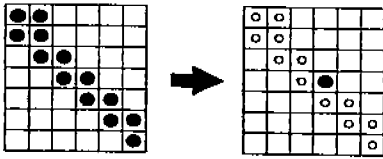


(그림 3) 제안된 대화식 클러스터링 기법의 흐름도
(Fig. 3) The proposed algorithm flow

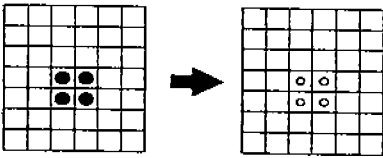
2.2 세션화

영상 데이터의 세션화를 통해 시스템의 처리 부하를 줄이는 동시에 벡터화에 적합한 형태로 변화하였다.

Zhang과 Suen의 알고리즘은 연결성을 보존하면서 빠른 속도로 세선화를 수행하지만, 굵기 2인 45° 각도의 대각선과 네 개의 점으로 이루어진 사각형에 대해서는 (그림 4)에 보인 바와 같이 형태를 보존하지 못하는 결점을 가지고 있다[6].



(a)



(b)

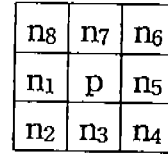
(그림 4) Zhang과 Suen의 알고리즘의 오류
(a) 굵기 2인 대각선의 세선화
(b) 4개의 점으로 이루어진 사각형의 세선화
(Fig. 4) Error of Zhang and Suen's algorithm
(a) Thining of diagonal line with thickness 2
(b) Thining of square consisting of 4 dots

(그림 4)에서 ●는 점, ○는 세선화 결과 제거된 점을 나타낸다. 본 논문에서는 세선화를 위해 Zhang과 Suen의 알고리즘의 결점을 해결한 Lü와 Wang의 알고리즘을 사용하였다[7].

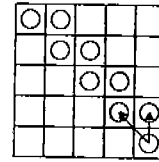
2.3 벡터화

세선화된 데이터의 점들을 추적하면서 벡터 데이터로의 변환을 수행한다. 영상의 좌상단에서 출발하여 만나는 첫 번째 점을 추적의 시작점으로 잡으며, 각 점 p의 8개의 이웃점은 (그림 5)와 같이 반시계 방향의 순서로 정의된다. Lü와 Wang의 알고리즘으로 세선화 했을 경우엔 (그림 6)과 같이 굵기가 2인 대각선이 결과 영상에 존재한다. 이 대각선을 추적해 나가면 같은 추적 방향으로 두 개의 이웃점이 존재하게 되어 문제가 된다.

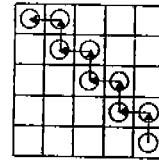
이를 해결하기 위해, 점 p의 이웃점의 개수를 식(1)



(그림 5) 점 p의 8개의 이웃점
(Fig. 5) 8 neighbors of point p



(그림 6) 굵기 2인 대각선
(Fig. 6) Diagonal line with thickness 2



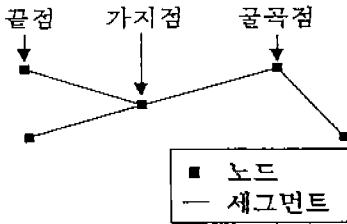
(그림 7) 굵기 2인 대각선의 추적
(Fig. 7) Tracing of diagonal line with thickness 2

과 같이 n_1, n_3, n_5, n_7 에 우선권을 주어 계산하면, 이 대각선의 추적은 (그림 7)과 같이 적절히 이루어진다.

$$\begin{aligned}
 (\text{이웃점의 개수}) &= n_1 + n_3 + n_5 + n_7 \\
 &+ \bar{n}_1 \wedge \bar{n}_3 \wedge n_2 + \bar{n}_3 \wedge \bar{n}_5 \wedge n_4 \\
 &+ \bar{n}_5 \wedge \bar{n}_7 \wedge n_6 + \bar{n}_7 \wedge \bar{n}_1 \wedge n_8
 \end{aligned} \quad (1)$$

이웃점의 개수가 1개이면 끝점(end point), 3개 이상이면 가지점(branch point)으로 간주하고, 개수가 2개이면 일반점(regular point)으로 간주한다. 동시에 Wall과 Danielsson의 직선 근사화 알고리즘을 적용하여, 꺾여진 일반점을 감지해 굴곡점(curve point)으로 간주한다[8]. 끝점과 가지점, 굴곡점을 노드로 정의하고, 노드와 노드를 연결한 근사화된 직선을 세그먼트로 정의하며, (그림 8)에 노드와 세그먼트를 나타내었다.

또한, 본 논문에서 사용된 Wall과 Danielsson의 직선 근사화 알고리즘을 정리하면 다음과 같다.



(그림 8) 세그먼트와 노드의 도해
(Fig. 8) Illustration of segment and node

(a) 추적 시작점이 원점이 되도록 좌표계를 이동한다.
 $f_i=0$ 으로 초기화한다.

(b) $\Delta x_i, \Delta y_i$ 를 계산하고 다음 점인 (x_i, y_i) 로 이동한 후, 다음 계산을 수행한다.

$$f_i = f_{i-1} + \Delta f_i,$$

여기서 $\Delta f_i = x_i \cdot \Delta y_i - y_i \cdot \Delta x_i$ (2)

시작점에서 (x_i, y_i) 까지의 현재 직선 세그먼트 길이 L_i 를 계산하고, 다음 조건을 검사한다.

$$|f_i| \leq T \cdot L_i \quad (3)$$

(c) 식 (3)의 조건을 만족하면, i 를 증가시키고 (b)에서

<표 1> 노드 테이블 구조
<Table 1> Structure of node table

번호	좌표	세그먼트 연결수	각 방향으로 연결된 세그먼트의 번호							
			1	2	3	4	5	6	7	8
170	(128, 311)	3	125	0	0	0	127	0	128	0
171	(70, 505)	1	0	128	0	0	0	0	0	0

<표 2> 세그먼트 테이블 구조
<Table 2> Structure of Segment table

번호	노드1	노드2	길이	각도(deg.)	형(type)	연결선 번호
127	167	170	17	175	문자	0
128	170	171	32	83	직선	97

부터 반복한다. 만족하지 않으면 (x_{i-1}, y_{i-1}) 를 굴곡점으로 기록하고, 그 위치를 시작점으로 하여 모든 점이 처리될 때까지 (a)에서부터 알고리즘을 반복한다.

식(3)의 T는 근사화 세그먼트의 단위 길이당 최대 허용 영역 편차(maximum allowed area deviation)이며, 값이 클수록 긴 세그먼트를 생성한다.

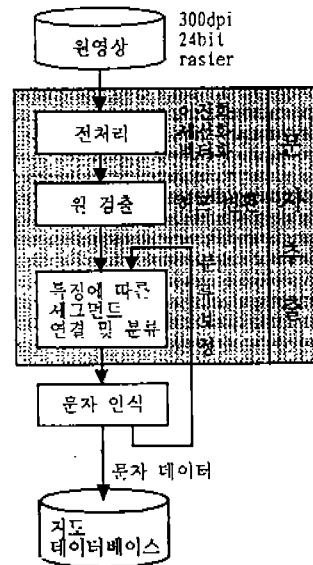
이때, 확인된 노드와 세그먼트들은 <표 1, 2>의 노드 테이블과 세그먼트 테이블에 각각 저장된다. 세그먼트 테이블의 형과 연결선 번호 항목의 내용은 다음 단계에서 결정된다.

3. 문자 추출 알고리즘

3.1 알고리즘 구성

입력된 지도 영상은 전처리 과정으로서 대화식 클러스터링 기법에 의한 이진화와 세선화를 거치고, 세선화된 영상을 벡터 데이터의 집합으로 변환하여 테이블에 저장한다. 제안된 알고리즘은 세그먼트의 길이 특징을 이용하여 각 세그먼트가 문자와 직선 중 어느 것에 포함되는 것인지를 판단한다.

여러 개의 짧은 세그먼트들로 이루어진 원은 문자에 속하게 되어 문자의 추출을 방해하기 때문에 직선



(그림 9) 문자 추출의 블록도
(Fig. 9) Block diagram of extraction of character

의 분리과정 전에 미리 허프 변환을 수행하여 분리시킨다.

각 직선에 대해서는 직선 전체의 길이가 고려되도록 해야 하므로 직선을 이루고 있는 세그먼트들을 서로 연결시킨다.

구성 요소의 분리를 위한 임계값은 직선 계층에 문자 요소가 섞이지 않도록 설정해야 한다. 문자 계층에 섞인 직선 성분은 문자 인식 단계에서 여과시킨 후 직선 계층에 포함시켜 연결 단계를 다시 거치게 한다.

(그림 9)는 문자 추출의 과정을 설명한 블록도이다.

3.2 원시 요소 분류

3.2.1 원

세션화된 영상에 대해 허프변환을 수행하여 원을 검출할 수 있다[3]. 모든 크기의 원을 찾기 위해서는 3차원 파라미터 공간을 사용해야 하지만 계산 시간을 줄이기 위해 반지름 파라미터에 몇 개의 특정 값들만을 적용한다. 세션화된 영상에 대해 원 함수의 허프 변환을 수행하면, 원의 중심점이 위치한 지점에 해당하는 셀을 중심으로 큰 값들이 분포한다. 이들 중 임계값 이상을 갖는 셀의 인접 집합들에 대해 각 개씩의 피크점(peak point)들만을 선택하여 그 점들을 원의 중심점들로 판단한다. 이때, 조건 1과 조건 2를 동시에 만족하는 세그먼트를 원으로 정한다.

조건 1

세그먼트의 양 끝점을 이루는 두 개의 노드로부터 원의 중심점까지의 거리가 L1, L2 일 때,

$$\begin{aligned} L1 &= (\text{반지름의 크기} \pm \alpha_{\text{circ}}) \\ \wedge L2 &= (\text{반지름의 크기} \pm \alpha_{\text{circ}}) \end{aligned} \quad (4)$$

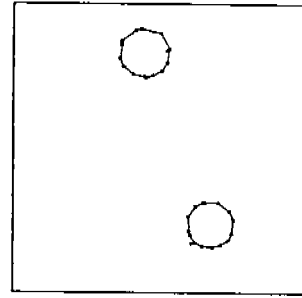
이며, 이때 α_{circ} 은 허용 오차이다.

조건 2

$$(\text{세그먼트의 길이}) < (\text{원의 반지름}) \times \frac{2}{3} \quad (5)$$

조건 2는 원을 가로지르는 세그먼트를 제외하기 위한 것이다. (그림 10)은 원으로 판단된 세그먼트들을 나타낸다. 아래쪽 원의 왼쪽 아래 부분에서 볼 수 있

듯이 원의 접선 방향으로 붙어 있는 작은 세그먼트의 경우는 원의 일부분으로 잘못 인식될 수가 있다.



(그림 10) 원으로 판단된 세그먼트
(Fig. 10) Segments decided as circle

3.2.2 직선

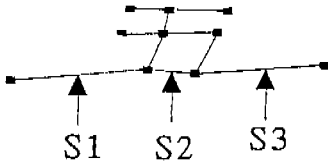
작성된 세그먼트 테이블의 길이 정보를 기준으로 문자와 직선의 분리를 시도하였다. 그러나 문자에 포함된 소수의 긴 세그먼트와, 직선에 포함된 짧은 세그먼트가 존재하기 때문에 임계값을 정하는 데에 어려움이 있다. 또한, (그림 11)과 같이 문자가 직선에 붙은 경우엔 직선이 몇 개의 세그먼트들로 나누어지게 된다. 이 문제를 해결하기 위해 각 세그먼트와 이웃하는 세그먼트와의 각도가 임계값 D_{line} 보다 작을 경우 서로 결합하여 연결선(linked line)을 만들었으며, 연결된 세그먼트들의 길이의 합을 연결선의 길이로 정했다. <표 3>은 연결선들의 정보를 저장한 연결선 테이블을 보이고 있다. 테이블에서 최소 세그먼트 번호와 최대 세그먼트 번호는 각 연결선이 포함하고 있는 세그먼트들의 번호 중 최소와 최대의 번호이다.

이때, 조건 3을 만족하는 세그먼트를 직선으로 정한다.

조건 3

$$\begin{aligned} (\text{세그먼트의 형}) &\neq \text{원} \\ \wedge (\text{세그먼트가 포함된 연결선의 길이}) &> L_{\text{line}} \end{aligned} \quad (6)$$

이때, L_{line} 은 임계값이다.



(그림 11) 직선에 붙은 문자의 예
(Fig. 11) Example of character attached to straight line

〈표 3〉 연결선 테이블
(Table 3) Linked line table

번호	길이	최소 세그먼트 번호	최대 세그먼트 번호
97	73	128	171
98	52	164	270
.....			

3.2.3 구획 안의 분할선

실험 대상으로 사용한 (그림 12)의 영상에서 볼 수 있듯이 지번약도의 각 구획을 분할하고 있는 직선들은 짧지만 대개 긴 직선에 수직으로 붙어 있다. 조건 4를 만족하는 세그먼트를 분할선(separated line)으로 정한다.

조건 4

- (세그먼트의 형) ≠ 원
- ∧ (세그먼트의 형) ≠ 직선
- ∧ (세그먼트가 포함된 연결선의 길이) > L_{sep}
- ∧ (이웃 세그먼트와의 각도) = 90° ± α_{sep} (7)

이때, L_{sep}은 임계값이고, α_{sep}은 허용 오차이다.

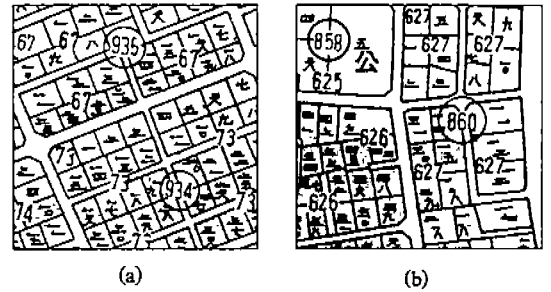
3.2.4 문자

원과 직선, 분할선들을 분리하고 난 후 남은 세그먼트를 문자(character)로 정한다.

4. 실험 및 검토

본 논문에서는 직선 요소의 비율이 높고 구성이 비교적 단순한 강남지역의 1:3,000 축척 지번약도(180

매중 71-2호)를 실험 대상으로 선택하였다. 지번약도는 직선들로 이루어진 구획(筆地)과 한자로 표기된 지번(地番), 소수의 지명, 그리고 원으로 둘러싸인 아라비아 숫자로 구성되어 있다.



(그림 12) 실험 대상으로 사용된 지번약도 영상
(a)실험 영상 1 (b)실험 영상 2
(Fig. 12) Register maps used as experiment target
(a)Test image 1 (b)Test image 2

입력 영상은 300dpi의 해상도, 24bit 칼라의 512×512 pixel 크기를 갖는다. (그림 12)는 사용된 지번약도 칼라 영상의 예이다.

실험을 위해 Pentium 100MHz CPU와 32MByte RAM을 내장한 시스템을 사용했으며, 프로그램은 C언어로 작성하였다.

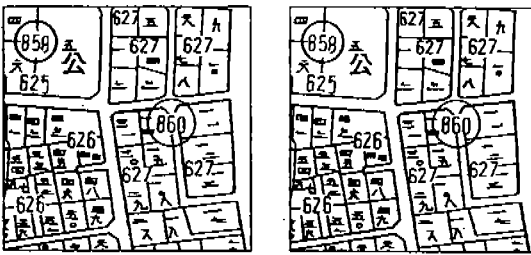
실험 영상을 대상으로 기존의 K-means 알고리즘과 제안된 알고리즘에 의한 클러스터링 수행 결과를 (그림 13)에 보였다. 두 가지 영상에 대해 흑색, 백색, 황색이 추출되기를 기대하였으나, K-means의 경우, 실험 영상 1은 원하는 색이 추출되었고, 실험 영상 2는 추출되지 않았으며, 제안된 알고리즘을 사용한 경우에는 실험 영상 1, 2 모두 원하는 색을 추출할 수 있었다. 이 결과를 (그림 13)에 보였다.

(그림 14, 15)는 실험 영상을 대화식 클러스터링 알고리즘으로 흑색만을 추출한 영상을 이용하여 문자영역을 추출한 결과이다. 이때 래스터 영상에서 연결성분만을 고려하여 문자를 추출한 결과와 제안된 문자추출 알고리즘을 적용하여 추출한 결과를 비교하였다.

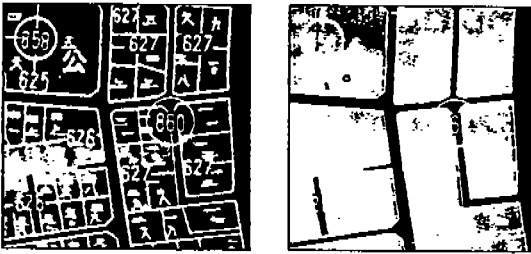
(그림 14(c))에서 'A'로 표시된 부분의 경우, 이웃 세그먼트와의 각이 임계값 이상이 되어 직선에 소속되지 않았으며, 구획안의 분할선을 검출하는 조건 4

에도 만족되지 않아 문자 영역으로 편입되었다.

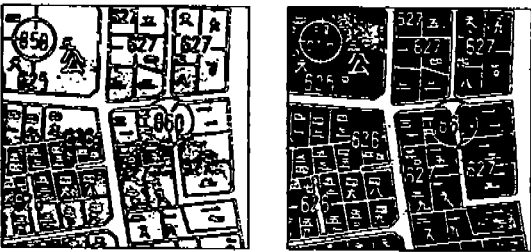
또한 (그림 15(d))에서 'B'로 표시된 부분의 경우, 원 주위의 직선 성분은 긴 직선에 연결되어 있지도 않고, 분할선 추출 조건에도 만족되지 않았으므로, 문자 영역에 포함되어 있음을 알 수 있다.



(a) 흑색(K-means) (b) 흑색(제안된 방법)



(c) 백색(K-means) (d) 백색(제안된 방법)



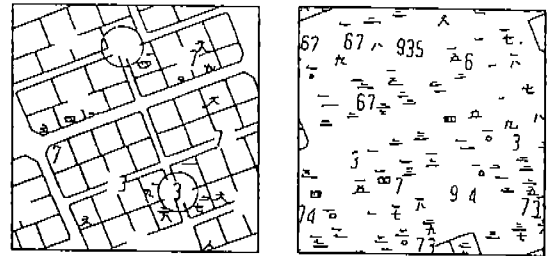
(e) 갈색(K-means) (f) 황색(제안된 방법)

(그림 13) 클러스터링 결과(실험 영상 2)

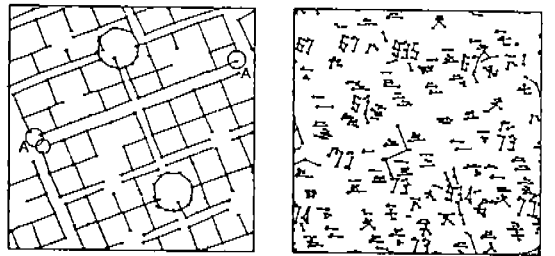
- (a),(c),(e) K-means 알고리즘 결과
- (b),(d),(f) 제안된 알고리즘 결과

(Fig. 13) A result of clustering(Test image 2)

- (a),(c),(e) K-means algorithm
- (b),(d),(f) by Proposed algorithm



(a) (b)



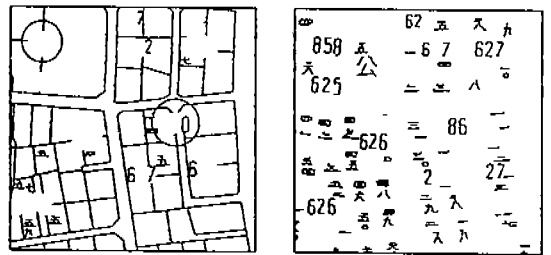
(c) (d)

(그림 14) 문자 추출의 예(실험 영상 1)

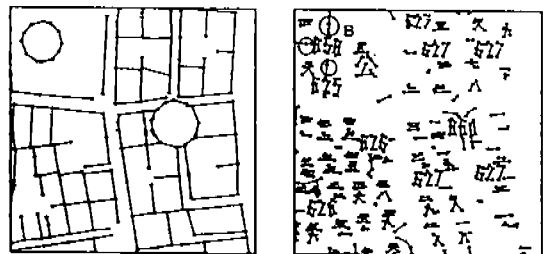
- (a) 연결 성분을 이용한 선분 추출
- (b) 연결 성분을 이용한 문자 추출
- (c) 제안된 알고리즘을 이용한 선분 추출
- (d) 제안된 알고리즘을 이용한 문자 추출

(Fig. 14) Example of character extraction(Test image 1)

- (a) extracted line segments using CC
- (b) extracted characters using CC
- (c) extracted line segment using suggested algorithm
- (d) separated character using suggested algorithm



(a) (b)



(c) (d)

- (그림 15) 문자 추출의 예(실험 영상 2)
- (a)연결 성분을 이용한 선분 추출
- (b)연결 성분을 이용한 문자 추출
- (c)제안된 알고리즘을 이용한 선분 추출
- (d)제안된 알고리즘을 이용한 문자 추출

(Fig. 15) Example of character extraction(Test image 2)
 (a)extracted line segments using CC
 (b)extracted characters using CC
 (c)extracted line segment using suggested algorithm
 (d)separated character using suggested algorithm

제안된 문자 추출 알고리즘에 의해 분리된 문자 계층을 보면, 영상의 가장자리에 문자로 오인된 선들이 많은 비중으로 존재함을 알 수 있다. 이 선들은 영상의 제한된 크기에 의해 잘려(clipping) 길이가 짧아진 것이므로, 주변 영상과 비교하여 다시 교정할 수 있다. 따라서 이러한 부분들은 문자의 인식과 같은 과정을 통하여 보정이 가능할 것으로 보여 진다.

본 논문에서 제안된 문자 추출 알고리즘의 경우에는 선과 접촉되어 있는 문자도 분리가 가능하므로 문자 인식의 측면에서는 유리한 점이 되지만, 세선화와 백터화 과정을 통한 문자 정보의 왜곡 및 문자 이외의 정보가 섞여 있는 점이 불리한 조건이 될 수 있다. 따라서 백터 정보를 이용하여 문자 영역의 상태 즉, 크기, 방향등과 같은 정보를 알아내고, 문자의 인식은 원 영상에서 실행하는 방법이 최적의 대안으로 생각 된다.

5. 결 론

지도 영상에 포함된 문자를 인식하기 위해서는 먼저 문자의 분리 과정이 필요하지만, 직선에 중첩된 문자의 경우 래스터 영상에서는 적절한 분리가 어렵다.

본 논문에서는 칼라 지도 영상의 색 정보를 대화식 클러스터링 알고리즘으로 색 별로 분리하여 문자 영역을 추출하였고, 문자가 포함된 영상을 백터화한 후 각 성분의 상호 관계를 조사하여 결합, 분류하는 알고리즘을 사용함으로써 문자와 직선과 원의 분리를 시도했다.

실험 결과 직선, 원, 문자 계층은 짧은 시간 내에 분리되었으나, 주변선과의 관계가 약한 소수의 직선들이 문자 계층에 포함되었다. 그러나 직선이나 원에 붙은 문자들은 독립된 연결 성분으로 분리되었으므로, 본래 의도했던 문자 인식의 전처리 과정으로는

충분하다고 판단된다. 문자 계층에 포함된 직선들은 단순한 형태를 이루고 있으므로 문자의 인식 과정에서 쉽게 여과될 수 있을 것이다.

참 고 문 헌

- [1] R. Kasturi et al., "Map Data Processing in Geographic Information System," Computer, pp.13-21, Dec. 1989.
- [2] S. Suzuki and T. Yamada, "MARIS:Map Recognition Input System," Pattern Recognition, Vol. 23, No. 8, pp.919-933, 1990.
- [3] R. C. Gonzalez and R. E. Woods, 'Digital Image Processing', Addison-Wesley Publishing Co. Inc., 1992.
- [4] L. A. Fletcher and R. Kasturi, "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images," IEEE trans. PAMI Vol. 10, No. 6, pp.910-918, 1988.
- [5] Sing-Tze Bow, 'Pattern Recognition and Image Preprocessing', Marcel Dekker, Inc., 1992.
- [6] T. Y. Zhang and C. Y. Suen, "A Fast Parallel Algorithm for Thinning Digital Patterns," Commun. ACM Vol. 27, No. 3, pp.236-239, Mar. 1984.
- [7] H. E. Lü and P. S. P. Wang, "A Comment on 'A Fast Parallel Algorithm for Thinning Digital Patterns'," Commun. ACM Vol. 29, No. 3, pp. 239-242, 1986.
- [8] K. Wall and P. E. Danielsson, "A Fast Sequential Method for Polygonal Approximation of Digitized Curves," CVGIP 28, pp.220-227, 1984.



안 창

1987년 단국대학교 전자공학과 (공학사)
 1989년 단국대학교 대학원 전자공학과(공학석사)
 1995년~현재 단국대학교 전자공학과 박사과정
 1989년 1월~1994년 12월 기아정

보시스템 연구소
 관심분야: 영상신호처리, GIS



박 찬 정

1966년 3월~1970년 2월 서울대학교 수학과(이학사)

1982년 2월~1984년 2월 단국대학교 전자공학과(공학석사)

1984년 3월~1991년 2월 단국대학교 전자공학과(공학박사)

1988년 3월~1990년 3월 강릉대학교 전자계산학과 전임강사

1990년 4월~1994년 3월 강릉대학교 전자계산학과 조교수

1994년 4월~현재 강릉대학교 전자계산학과 부교수

1993년 3월~현재 강릉대학교 전자계산소 소장

관심분야: 컴퓨터 시스템 분산 및 병렬처리 멀티미디어



이 상 범

1974년 연세대학교 전자공학과(공학사)

1978년 서울대학교 대학원 전자공학과(공학석사)

1986년 연세대학교 대학원 전자공학과(공학박사)

1984년 미국 IOWA대학교 컴퓨터공학과 객원교수

1979년~1992년 단국대학교 전자공학과 교수

1993년~현재 단국대학교 컴퓨터공학과 교수

관심분야: 컴퓨터구조, 영상신호처리, GIS