

자동 벡터링 도구의 설계 및 구현

이상훈*, 임은정*, 허봉식**, 김민환***

*부산대학교 지형정보협동과정

**동의공업대학 영상정보처리학과

***부산대학교 컴퓨터공학과

Design and Implementation of an Automatic Vectoring Tool

Sang-Hoon Lee*, Eun-Jung Lim*, Pong-Sik Ho**, Min-Hwan Kim***

*Dept. of Geographic Information System, Pusan national university

**Dept. of Image Processing, Dongeui Institute of Technology

***Dept. of Computer Engineering, Pusan national university

요 약

자동 벡터링 도구는 스캐닝된 도면 영상으로부터 벡터 데이터를 자동으로 생성시키기 위해 사용된다. 기존의 벡터링 도구에서는 벡터링 작업도중 과도한 사용자 개입을 요구하고 있어 생산성 및 정확성이 저하되는 문제점을 가지고 있다. 이를 효과적으로 해결하기 위해, 본 논문에서는 스캐닝된 도면 영상에서 자동화에 문제가 될 수 있는 부분을 찾아 제거하기 위한 전처리 방법과 벡터링 대상물의 기하학적 특성을 잘 반영해 줄 수 있는 자동 선추적 방법, 사용자 편의 위주의 벡터 데이터 편집 및 수정 작업을 위한 후처리 방법을 이용해 자동 벡터링 도구를 설계하였다. 또한, 입력되는 도면 영상 중 과도한 전처리가 요구되는 영상은 사용자가 직접 화면을 보면서 디지털링할 수 있는 화면 디지털링 방법도 포함시켰다. 실험을 통해 구현된 자동 벡터링 도구가 기존의 반자동 벡터링 도구에 비해 효율성 및 정확성이 높다는 것을 확인하였다.

1. 서론

본 논문에서는 멀티미디어 산업의 성장과 더불어 수요가 급증할 것으로 기대되는 GIS의 구축에 유용하게 사용될 수 있는 벡터링 도구를 설계하고 구현하였다.

기존 도면을 이용해 지형정보를 구축하는 방법으로는 주로 디지털라이저를 이용한 수동식 디지털라이징 방법을 사용할 수 있었다. 그러나, 이 방법은 벡터링

과정에서 사용자가 벡터링 대상물들을 찾아가면서 하나씩 지정해야 하고 작업 후 많은 수정작업이 필요하기 때문에 전체 벡터링 작업의 효율성 및 정확성이 저하되고 값비싼 디지털라이저를 구비해야 하는 부담이 있다[1].

한편, 기존의 상용 벡터링 도구들은 양질의 원판 지도나 도면에서와 같이 매우 깨끗하게 나타나는 벡터링 대상물에는 적합하지만 잡영 및 질 저하 현상

이 심하게 나타나는 대상물에는 효과적으로 적용할 수 없다. 또한, 벡터링 과정에서 대상물의 기하학적인 특성을 고려하지 않기 때문에 벡터링 과정 및 작업 후에 너무 많은 사용자 개입을 요구하는 경향이 있어 수동식 디지털라이징 방법에서 제기된 문제를 효과적으로 해결하지 못하고 있다[2].

이에, 본 논문에서는 벡터링 작업의 생산성 및 효율성을 향상시키기 위해 현재 값싸게 공급되고 있는 스캐너를 이용해 대화형으로 지형정보를 구축하되 최대한 자동화된 벡터링 도구를 개발하였다. 특히, 잡영 및 질 저하 현상이 심한 대상물에도 잘 적용할 수 있으면서 대상물의 기하학적인 특성을 자동으로 인식하여 처리할 수 있도록 설계하여 개발하였다.

2장에서는 구현되어질 자동 벡터링 도구의 전체적인 설계와 세부 기능들에 대한 필요성 및 방법론에 대해 기술하고, 3장에서는 구현될 결과에 대해 논한다.

2. 자동 벡터링 도구 설계

2.1. 스캐닝된 도면 영상의 전처리

기존의 자동 벡터링 도구들은 도면 영상에 벡터링 대상물들과 함께 포함되어 있는 문자나 심볼, 잡영에 대한 아무런 처리 없이 자동 벡터링 방법을 적용시킨다. 이 방법으로 생성한 벡터 데이터는 많은 오류를 포함하고 있어 많은 수정작업이 요구된다. 이 경우, 벡터 데이터를 사용자가 직접 보면서 수정해야 하기 때문에 오류 수정 대상이 되는 영역을 찾고 수정하는데 과도한 수작업이 요구된다. 이에, 본 논문에서는 벡터링 작업 후 사용자의 개입 요인과 벡터링 작업의 자동화에 문제가 되는 요인을 제거하기 위한 전처리 방법을 사용하였다. 아울러, 사용자의 개입을 전처리 단계에 집중시켜 전체 벡터링 작업의 효율성을 증가시켰다.

스캐닝된 이진 도면 영상을 보면 종이 지도의 팽창이나 수축 및 스캐닝시 발생하는 광학적인 오차로 인해 모양이 뒤틀려진 경우가 있다. 이에, 본 논문에서는 여러 종류의 영상 warping 방법들[3,4] 중에서 bilinear warping 방법을 사용하였다.

도면 영상에는 벡터링 대상물과 불필요한 요소들이 함께 존재하는데 이러한 불필요한 요소들을 자동으로 검출하고 제거하기 위해, 우선 전체 영상을 조사하여 8-방향으로 연결된 연결요소(connected component)를 구한다[5]. 문자, 기호, 잡영들은 비교

적 크기가 작은 특징이 있으므로 구해진 연결 요소들 중에서 연결 요소를 구성하는 픽셀의 개수가 사용자가 정한 임계값보다 작은 연결 요소들을 제거하게 된다. 그런데 이때 적절하지 못한 임계값 설정으로 의미있는 요소들이 제거될 수 있으므로, 이를 방지하기 위하여 현재 설정된 임계값으로 제거될 모든 요소들을 사용자가 차례대로 볼 수 있고 특정 요소를 제거 혹은 제거시키지 않게 할 수 있는 방법을 사용한다. 그러나 이 방법에서는 단순히 픽셀의 개수 정보만을 이용하기 때문에, 벡터링 대상물과 의미 없는 요소들이 겹쳐있는 경우에는 적용할 수 없다. 이런 경우, 미리 선추적을 행하여 벡터링 대상물에 겹쳐있는 의미없는 요소들을 찾아내어 제거해야 하고, 자동 선추적 방법으로 처리 불가능한 지역을 자동으로 검출해서 사용자의 개입을 요구해야 한다. 이를 위해 선추적시 여러 가지 조건을 두어 이들 조건 중 하나라도 만족하지 않는 대상물들에 대해서는 즉시 사용자의 개입을 요구하도록 하여 문제를 해결하였다[6]. 이 과정에서 행해지는 도면 영상에 대한 사용자의 수정작업으로 인해 벡터링 대상물에 불필요한 짧은 가지가 남아 있을 수 있다. 이런 불필요한 부분은 잡영까지 제거 알고리즘을 사용하여 제거하였다[7].

스캐닝된 도면 영상에 대한 전처리 작업은 자동 벡터링 과정에서 문제가 발생할 수 있는 요인을 제거하는데 목적이 있고 벡터링 작업의 자동화를 가능케 하는 작업이다. 이에, 본 논문에서는 전처리 작업의 정확성과 벡터링 작업의 완전 자동화를 보장하여 사용자의 개입을 전처리 단계에 집중시키기 위해 전처리된 도면 영상을 검증하는 방법을 사용하였다.

2.2. 자동 선추적

스캐닝된 도면 영상으로부터 디지털라이징을 행하는 방법은 주로 벡터링 대상물의 현재 위치로부터 일정거리 앞까지의 구성픽셀들을 모두 살펴보아 진행여부를 결정하는 방법을 사용하고 있다. 이러한 방법은 매우 간단한 구조로 구성되어 있는 부분에서도 불필요하게 많은 연산을 행해야 하므로 전체적인 성능의 저하를 야기시키는 경향이 있다. 또한, 기존의 자동 벡터링 도구에서는 벡터링 대상물의 기하학적인 특성을 자동으로 인식하여 처리할 수 있는 기능이 없기 때문에 교차점이나 코너부분을 정확하게 벡터링하지 못하고 있다.

그림 1은 미국 Able Software사에서 개발한

R2V를 대상으로 실험한 결과이다. 교차점을 자동으로 검출하고 인식하지 못하기 때문에 교차로에서 교차점이 2개(a 교차로)로 잘못 선정되거나 교차점이 선정되지 못하는(b 교차로) 현상이 발생한다.

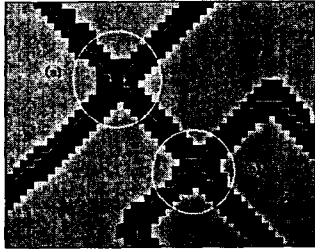


그림 1. R2V 실험 결과

그림 2은 스위스 CyberSonic Technologies사에서 개발한 ScanVector를 대상으로 실험한 결과이다. 급격히 변화되는 코너부분에 대한 벡터 데이터가 실제 도면을 벗어나 있음을 보여주고 있다.

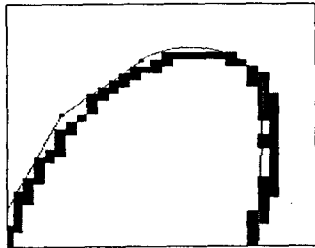


그림 2. ScanVector 실험 결과

이와 같이, 대부분의 자동 벡터링 도구들은 발생된 오류를 수정하기 위해 사용자의 과도한 후처리 작업을 요구하고 있다.

이에, 본 논문에서는 벡터링 대상물에서 벡터링할 부분의 기하학적 복잡도를 고려해 적절한 계산량을 선택적으로 사용하는 일종의 적응적 선추적 방법 [8]을 사용한다. 이 방법의 기본은 그림 3에서와 같이 선추적 조사점을 중심으로 선추적 진행 방향으로 적절한 반지름의 원을 그려 원과 벡터링 대상물이 중첩되는 부분(이하 뷰라 함)을 이용해서 다음 진행 방향 및 대상물의 기하학적 특성을 판별하는 방법이다.

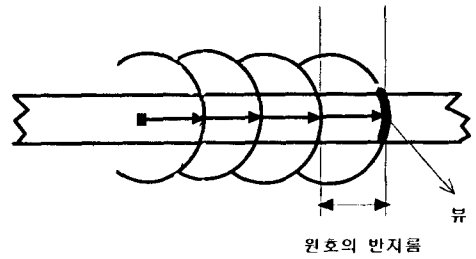


그림 3. 선추적 방법

선추적 진행 방향이 급격하게 변화하는 코너일 경우 그림 4에서와 같이 코너인 부분을 자동 인식하여 코너 부분의 굴곡을 잘 표현할 수 있도록 해당 영역을 세선화 한다.

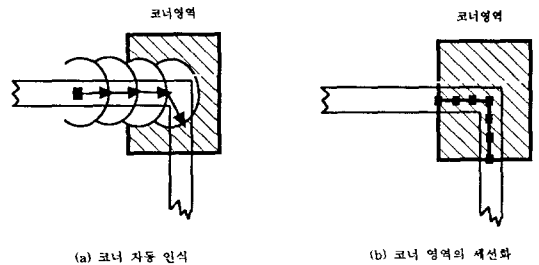


그림 4. 코너의 자동 인식 및 처리 방법

또한, 선추적시 뷰의 개수가 여러개 나오는 교차지점인 경우 그림 5에서와 같이 자동으로 교차점을 선정하고 교차점에 해당되는 갈래길은 자동으로 선정된 교차점에서 그린 원과 갈래길이 중첩되는 부분을 이용해서 등록, 처리한다.

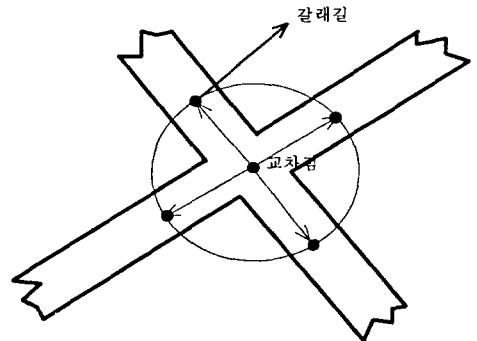


그림 5. 교차점의 자동 선정 및 갈래길 처리 방법

그림 6은 앞에서 설명한 방법들을 이용하여 실제 자동 벡터링 단계에서 사용될 자동 선추적 방법을 설계하여 도식화한 것이다.

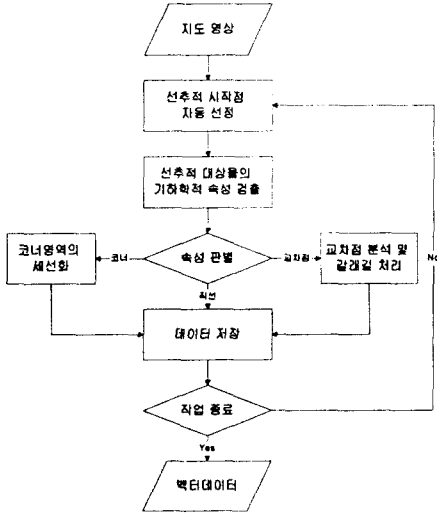


그림 6. 자동 선추적 방법 흐름도

2.3. 화면 디지털라이징

스캐닝된 도면 영상들 중에는 여러 레이어가 중첩되어 있거나 많은 잡영이 포함되어 있어, 전처리에 너무 과중한 노력이 필요한 경우가 있다. 또한, 점선이나 철도와 같은 지도 요소들은 현재의 선추적 방법으로는 벡터링할 수 없는 영역들이다. 따라서 이런 영역을 벡터링 하기 위해 화면에 디스플레이된 스캐닝 지도를 사용자가 눈으로 확인하면서 마우스로 디지털라이징하는 화면 디지털라이징 방법이 필요하다.

일반적으로, 도면 영상에 있어서 화면 디지털라이징이 필요한 영역은 그림 7과 같이 입력 도면 전체가 아닌 특정 일부분일 경우가 많다. 그러므로 본 논문에서는 입력 도면 전체에 대하여 화면 디지털라이징으로 입력시킬 수 있는 방법 뿐 아니라, 입력 도면에서 자동화가 가능한 부분은 자동화시키고, 그렇지 못한 부분에 대해서는 화면 디지털라이저를 이용하여 수동으로 입력할 수 있는 방법을 동시에 제공한다. 사용자는 먼저 화면 디지털라이징으로 작업할 영역에 대해 영역 설정을 한다. 영역 설정 시 경계 부분에서 발생 할 수 있는 여러 가지 문제들을 피하기 위해 원을 이용하여 영역을 설정한다. 하지만 원 하나로 영역을 설정하게 되면 길쭉한 지역을 설정하기 위해서 자동화가 가능한 영역도 포함해야 하는 문제

가 발생하게 된다. 이에, 본 논문에서는 여러 개의 교차된 원을 하나의 영역으로 정의함으로써 영역 설정을 좀 더 자유롭게 할 수 있도록 하였다. 설정된 영역 외부는 앞에서 제시한 자동화 방법으로 벡터 데이터를 생성하고, 영역 내부는 화면 디지털라이징으로 벡터 데이터를 생성한다. 이러한 방법으로 생성된 두 벡터 데이터는 병합되어 하나의 완성된 벡터 데이터로 출력된다.

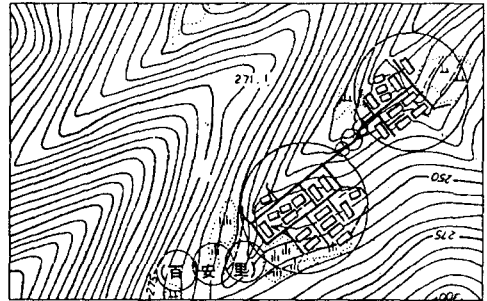


그림 7. 화면 디지털라이징 대상 영역 설정

화면 디지털라이징은 스캐닝된 도면 영상을 보면서 벡터링 할 대상물을 사용자가 마우스로 선택하면 선추적을 시작한다. 선 추적 도중 사용자 개입이 필요한 지점에 도달하게 되면 선 추적은 일시 중단되고, 이 영역을 확대한 영상이 디스플레이 된다. 이때 사용자는 개입이 필요한 원인을 분석하여 필요한 작업을 수행한다. 즉, 추적하고 있는 대상물이 다른 레이어에 의해 중첩되어 있거나 혹은 잡영이 있는 경우 진행할 방향으로 노드를 추가하고, 교차로인 경우에는 교차지점과 갈래길들을 지정한다. 또한 벡터 대상물의 끝에 도달한 경우에는 끝점을 등록한다. 화면 디지털라이징 작업은 하나의 연결요소를 단위로 작업을 하게 되며 하나의 연결요소에 대한 작업이 끝날 때마다 그 연결 요소에 대한 벡터 데이터가 만들어진다.

2.4. 후처리

스캐닝된 도면 영상을 바탕으로 생성된 벡터 데이터에는 입력된 도면 자체에 포함된 오류나 벡터링 과정에서 발생한 오류가 존재할 수 있으며 이러한 오류를 검출하고 수정하는 후처리 과정이 필요하다.

이때 사용자가 전체 도면을 조사하면서 벡터 데이터의 오류를 찾아 수정하는 것은 비효율적인 뿐만 아니라 일부 지역은 빠뜨리고 처리하지 않는 경우가 발생할 수도 있다. 그러므로 작업효율의 향상을 위해 후처리 작업이 필요한 지역을 자동으로 검출하여

처리할 수 있는 기능이 요구된다. 이에, 본 논문에서는 교차점과 코너 혹은 끝점을 자동으로 찾아 화면에 표시하고 사용자가 오류 여부를 판단하여 위치수정, 단선연결, undershoot, overshoot의 수정을 자동으로 수행하도록 한다. 또한 입력도면과 벡터 데이터를 중첩 시켜 보여줌으로써 사용자가 벡터 데이터의 오류를 쉽게 판별하여 작업할 수 있는 편리한 작업환경을 제공한다. 아울러 자동 검출 및 수정이 불가능한 부분에 대해서는 사용자가 노드(node), 버텍스(vertex), 아크(arc)를 임의로 편집할 수 있게 한다.

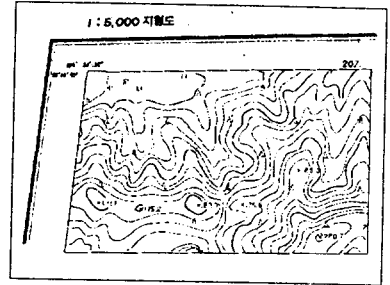
방대한 데이터를 다루는 지리정보시스템의 경우 사용되는 메모리의 양을 최소화하기 위해서 도면에 따라 벡터 데이터를 일반화하여 표현할 필요가 있다 [10]. 본 논문에서는 아크에 포함되어 있는 각 점에 속성값을 부여하고 사용자가 시각적 확인을 거쳐 할당한 임계값 이하의 속성값을 가지는 불필요한 점들을 제거함으로써 벡터 데이터를 일반화하여 표현한다. 다양한 특성을 가지는 아크들이 존재하는 도면에서는 한 도면에 동일한 임계값을 주는 것 보다 아크의 특성에 따라 아크 전체나 아크 내 임의 구간에서 선택적으로 임계값을 부여하여 일반화의 정도를 다르게 수행할 수 있다.

본 논문에서 획득한 벡터 데이터는 국립지리원 수치지도 표준 포맷인 DXF(Drawing Interchange File)[9]로 저장되도록 하였다.

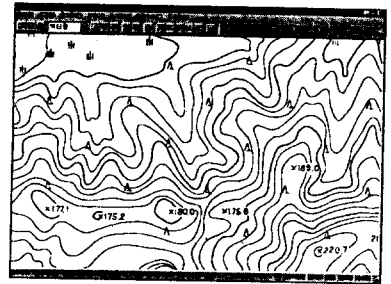
3. 자동 벡터링 도구의 구현

본 논문에서 설계한 자동 벡터링 도구는 일반 사용자들도 쉽게 사용할 수 있도록 Window95환경에서 Visual C++(ver5.0)을 이용하여 구현하였으며 스캐닝 해상도는 건설부의 수치지도 작성기준에서 제시된 400dpi를 사용하였다. 그림 8은 스캐닝된 입력 영상으로부터 벡터 데이터를 생성시키는 단계를 나타내고 있다. 그림 8(a)는 온도나 습도 혹은 스캐닝 시 광학적 오차로 인해 발생하는 도면 영상의 왜곡 현상을 나타내고 있으며 그림 8(b)에서는 이런 도면 영상의 왜곡을 bilinear warping방법을 이용하여 보정한 결과를 나타내고 있다. 그림 8(c)는 그림 8(b)에 나타난 의미없는 고립 요소와 벡터링 대상물과 겹쳐있었던 요소들을 제거한 결과를 나타내고 있다.

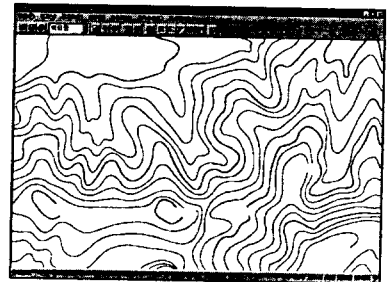
이렇게 전처리된 도면 영상은 자동 선추적 방법에 의해 자동 벡터링이 가능하며 선추적 작업에 의해 그림 8(d)과 같은 형태의 벡터 데이터가 생성된다.



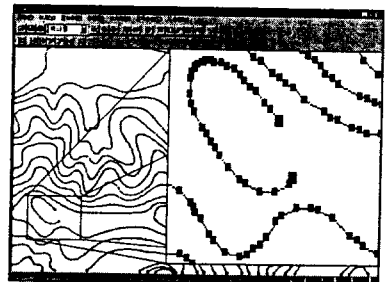
(a)



(b)



(c)



(d)

그림 8. 실행 결과

표 1은 본 논문에서 구현한 자동 벡터링 방법과 기존의 벡터링 방법을 국립 지리원 발행 원도와 VTRAK 데모용 영상을 사용하여 비교한 것이다. 이 비교를 통해 기존의 벡터링 방법은 도면의 복잡도에 따라 전체 벡터링 작업 수행시간의 차이가 매우 큰 반면, 제안된 방법은 도면의 종류와 관계없이 수행 시간의 차이가 크지 않음을 알 수 있다.

표 1 수행 시간 비교 (단위: 분)

	제안된 방법			기존의 방법
	전처리	선택적	합계	
지리원 발행 원도	5.85	0.87	6.72	16.17
VTRAK 데모 영상	5.18	2.15	7.33	34.70

4. 결론

본 논문에서는, 스캐닝된 도면 영상에 대한 자동 벡터링 방법들을 제안하고 이들을 이용해 자동 벡터링 도구를 구현하였다. 기존의 반자동 벡터링 도구에서는 벡터링 대상물들의 기하학적인 특성을 고려하지 않기 때문에 실제 도면 영상과 벡터 데이터간에 많은 오차를 포함하고 있다. 반면에, 본 논문에서 구현한 자동 벡터링 도구는 기존의 벡터링 도구와는 달리 벡터링 대상물들의 기하학적인 특성에 아주 민감하도록 설계되어 있기 때문에 정확한 벡터 데이터를 생성할 수 있어 후처리 과정에서의 사용자 개입을 대폭 감소 시켰다.

본 벡터링 도구는, 지도 영상이나 설계 도면 영상을 기반으로 벡터 데이터를 생성시키는 분야에서 효과적으로 활용될 수 있다.

향후, 보다 높은 처리 속도와 실제 좌표 지정 및 변환기능의 추가, 문자 및 심볼 처리 기능 추가를 위한 방법에 대해 연구할 것이다.

참 고 문 헌

[1] 박기석, GIS 지리정보시스템, 동서출판사, 1995
 [2] Tol Bernhardsen, Geographic Information Systems, Viak IT, 1992

[3] George Wolberg, Digital Image Warping, IEEE Computer Society Press, 1990
 [4] Randy Crane, A simplified approach to Image Processing, Prentice Hall, 1997
 [5] Gonzalez, Woods, Digital Image Processing, Addison Wesley, 1992
 [6] 장순형, 김종민, 김민환, 자동 벡터링을 위한 효율적인 전처리 기법 및 일괄처리 방법, 정보처리학회 학술발표회 논문집, 제4권 2호, pp. 1169-1174, 1997.10
 [7] 권태균, 김종민, 김성영, 김민환, 2x2 마스크를 이용한 윤곽선 추출 알고리즘의 응용에 관한 연구, 정보처리학회 학술발표회 논문집, 제4권 2호, pp. 1163-1168, 1997.10
 [8] 정기중, 김민환, 맹인식 선택적 및 교차점 자동 분석 방법에 의한 곡선 및 직선 벡터링 기법, 신호처리합동학술대회 논문집 Vol. 9, Part 2, pp. 767-770, 1996
 [9] Autodesk Inc, Standard Data Format DXF, 1988
 [10] 김주완, 김민환, 벡터 데이터의 다단계 표현에 관한 연구, 한국정보과학회 학술발표논문집, Vol. 21, No. 1, pp. 723-726, 1994