

# 등고선을 이용한 지형정보의 3차원 시각화 기법

## A 3-D Visualization Method for Geographical Information based on Contour Lines

한정규\*, 백중환\*\*, 황수찬\*\*

Jung-Kyu Han\*, Joong-Hwan Baek\*\*, and Soo-Chan Hwang\*\*

### 요 약

기존의 위성 영상이나 지도 이미지를 이용한 지형의 3차원 시각화 방법은 전처리 과정이 복잡하고 3차원 지형의 생성에 필요한 데이터가 방대하다는 단점이 있다. 따라서 본 논문에서는 수치지도를 이용하여 등고선 정보를 획득하고 이를 바탕으로 3차원 지형을 자동으로 생성할 수 있는 지형의 3차원 시각화 방법을 제시하고자 한다. 또한 지형 데이터의 3차원 시각화를 위해 반드시 처리해야 하는 세가지 문제인 등고선 대응 문제, 타일링 문제, 지맥 문제를 효율적으로 처리할 수 있는 방법을 소개한다. 이를 위해 가상 등고선의 개념을 정의하고 왜곡된 등고선을 대응되는 등고선과 유사한 모양이 되도록 확장하여 3차원 표면을 생성할 수 있도록 한다. 이 방법은 3차원 지형을 정확하고 세밀하게 표현할 수 있는 장점이 있다.

### Abstract

The existing visualization methods using the satellite images or map images require complicated preprocessing stages and a large amount of visual data to represent the 3-D terrain. This paper presents a 3-D visualization method for geographical information, which enables automatic generation of 3-D terrain. It is generated based on contour information obtained from a numerical map. This paper also introduces a method that resolves the three main problems needed to visualize 3-D terrain from contour lines such as correspondence, tiling, and branching. The virtual contour line is defined to extend a distorted contour line to have a similar shape to the corresponding contour line that is used to generate 3-D surfaces. It helps that 3-D terrain is represented exactly and in detail.

### I. 서 론

지리정보시스템(geographical information system)은 지리 및 지형에 관련된 자료를 수집, 저장, 관리하며, 적절한 처리를 가하여 원하는 형태로 출력할

수 있는 시스템이다. 현재 지리정보시스템은 도시 계획, 자원관리, 환경 관리, 재해정보 관리, 마케팅, 토지관리, 군사정보 시스템 등 다양한 응용분야에 활용되고 있으나 2차원의 평면적인 지리정보만으로는 사용자에 대한 정보 전달능력이나 그 응용분야에 많은 한계성을 갖는다. 특히 지리정보를 이용하는

본 연구는 한국과학재단의 특정기초과제(98-0102-06-01-3)와 인터넷 정보검색 연구센터의 연구비 지원에 의한 것임.

\* 한국항공대학교 컴퓨터공학과(Dept. of Computer Eng., Hankuk Aviation Univ.)

\*\* 한국항공대학교 전자·정보통신·컴퓨터공학부(The Schools of Electronics, Telecommunication, and Computer Engineering, Hankuk Aviation Univ.)

논문번호 : 2001-2-2

접수일자 : 2001년 6월 12일

각종 시뮬레이터 및 가상현실 시스템의 응용 분야가 증가함에 따라 실제 지형의 정확한 표현을 위한 지형의 3차원 시각화에 대한 요구가 더욱 높아지고 있는 실정이다[1].

3차원 지형의 시각화는 컴퓨터 그래픽스, 데이터베이스 기술, 공간 객체 모델링 등의 기술이 통합적으로 요구된다고 할 수 있다. 현재 지형의 3차원 시각화는 그래픽 도구를 이용하여 가상의 이미지를 생성하는 방법[2]과 실측 지형 데이터를 사용하는 방법[3], 두 방법의 혼합형[4] 등으로 나눌 수 있다. 가상의 이미지를 생성시키는 방법은 다양한 효과를 장면에 줄 수 있다는 장점이 있으나 이미지를 만드는데 시간과 비용이 많이 들며, 실제 지형을 정확히 3차원으로 표현하기 힘들다는 단점이 있다.

실측 지형 데이터를 이용하는 방법은 지도를 기본으로 3차원 지형을 생성한다는 측면에서 정확한 3차원 지형을 재현할 수 있다는 장점을 갖는다. 이 방법에서 사용되는 데이터는 일반적으로 정규격자망(regular grid network) 형태로 구성된다. 이러한 래스터 방식의 방법은 등고선 지도에서 특징점들을 검출하여 등고선들의 사이를 보간하는 방법으로 지형을 표현한다. 이 방법은 위상 영상이나 이미지 매칭 기법들을 이용하여 데이터 획득과 저장 과정을 자동화하기가 용이하다. 그러나 정규격자형 방식은 지형 형태에 관계없이 일정한 간격으로 데이터를 구축하므로 많은 기억용량을 필요로 하며, 지형의 특성을 제대로 반영하지 못한다는 단점이 있다[5]. 정규격자형 방식에서 데이터의 정확도는 추출된 데이터의 양에 비례한다.

벡터 방식으로 시각화를 하는 방법은 래스터방식보다 적은 데이터를 이용하여 지표면을 나타낼 수 있는 비정규삼각망(Triangulated Irregular Network, TIN) 모델을 들 수 있다. 구축방법이 정규격자망 모델보다는 어렵지만 기억장소를 절약할 수 있다는 장점때문에 ARC/INFO와 같은 상용 지리정보시스템에 많이 이용되고 있다[6]. 그러나 지도 도면의 이미지 프로세싱을 통해 데이터를 획득하므로 지도 도면에서 특징점을 추출하기 위한 전처리 과정이 필요하며 이 과정에 많은 시간과 비용이 든다[7]. 특히 등고선이 심하게 왜곡된 지형은 적절한 3차원 지형

을 생성하지 못하는 문제도 있다.

따라서 본 논문에서는 수치지도를 이용하여 등고선 정보를 획득하고 이를 바탕으로 3차원 지형을 자동으로 생성할 수 있는 지형의 3차원 시각화 방법을 제시하고자 한다. 특히 본 논문에서 제시한 시각화 알고리즘은 왜곡이 심한 지형도 가상의 등고선을 유도하여 정확하면서도 현실감 있는 3차원 지형을 생성할 수 있도록 하는 특징을 가지고 있다.

본 논문의 구성은 다음과 같다. 2장에서 지형 시각화를 위한 문제점과 관련 연구에 대해 설명한다. 3장에서는 2장의 문제들을 해결하기 위한 구체적인 등고선 확장 기법과 시각화 기법을 소개하고 4장에서는 구현 알고리즘과 실제 구현 결과를 설명하며 5장에서 결론을 맺는다.

## II. 관련 연구

지리정보는 다른 정보와는 달리 모든 정보가 공간상의 위치로 정의되고 분석된다. 따라서 효율적인 수치지형모델(Digital Terrain Model: DTM)의 구축은 지리정보시스템의 선결 조건이라 할 수 있다[8]. 일반적으로 지리정보시스템을 위한 수치표고모델(Digital Elevation Model: DEM) 데이터는 정규격자망 형태로 구성된다. 그러나 정규격자형 DEM은 지형 형태에 관계없이 일정한 간격으로 데이터를 구축하므로 많은 기억용량을 필요로 하며, 지형의 특성을 제대로 반영하지 못한다는 단점이 있다[5]. 정규격자형 DEM에서 데이터의 정확도는 추출된 데이터의 양에 비례하며 이는 정규격자형 데이터가 영구적인 저장방법으로 부적절함을 의미한다. 따라서 벡터 데이터를 이용한 지형의 DEM 2차원 지도의 등고선으로부터의 3차원 지형을 복원하는 방법은 여러 가지가 있다. 먼저 가장 많이 사용되는 방법으로 2차원 상에 임의로 분포되어있는 점들의 높이정보를 이용하여 가로/세로 등간격으로 분포되어 있는 격자점들의 높이를 보간하는 기법이 있다[4]. [9]에서는 격자선분과 등고선이 만나는 점의 높이정보로부터 스프라인을 구하고 이 스프라인의 격자점 높이를 계산하는 과정을 반복함으로써 모든 격자점에서의 높이를 보간하는 방법을 제시하였다. [8]

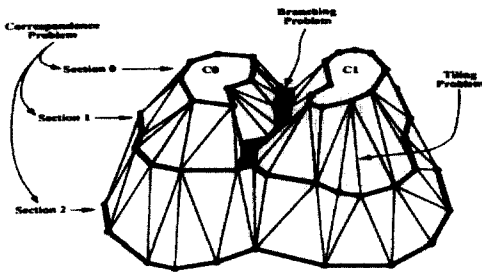


그림 1. 지형 시각화의 세 가지 문제  
Fig. 1. Three problems in terrain visualization.

에서는 적은 데이터를 이용하여 지표면을 나타낼 수 있는 TIN 방법을 제시하였다. TIN 구조는 지형의 형상에 따라 추출되어진 기하학적으로 랜덤한 데이터에 의하여 DEM을 구성하는 것으로 그 구축방법이 격자형 데이터 구조보다 어렵지만 기억장소를 절약할 수 있다는 장점을 갖는다. 이 이외에도 격자선 분과 등고선이 만나는 점의 높이를 직선으로 이어선형 보간하는 방법과 삼각분할 기법을 이용하여 스프라인 곡면으로 보간하는 방법[10] 등이 있다.

등고선을 이용하여 지형을 시각화하기 위해서는 그림 1에서와 같이 등고선으로부터 면을 구성할 때 발생하는 등고선 대응(correspondence)문제, 타일링(tiling) 문제, 지맥(branching) 문제를 해결해야 한다[11].

먼저 등고선 대응 문제는 등고선들간의 인접 관계를 결정하는 문제로서 등고선의 집합을 하나의 트리로 구성한다고 가정할 때, 각 등고선들간의 부모 자식관계, 형제 관계를 결정함을 의미한다. 대부분 등고선 대응 문제는 등고선간의 포함관계를 이용하여 해결한다.

다음, 타일링 문제는 등고선으로부터 표면을 구성하는 문제로서 일단 인접관계가 결정된 두 개의 등고선사이를 면으로 보간하는 문제이다. 일반적으로 두 등고선 상의 점들로 연속적인 삼각형을 구성하게 되는데 그 방법으로는 생성되는 객체의 전체부피를 최소화하는 방법[12]과 면의 넓이를 최소화하는 방법[13]이 있다. 부피 최소화 방법은 오목한 객체를 생성할 수 없다는 단점이 있으며, 면의 넓이를 최소화하는 방법은 잘못된 지형을 생성할 수 있다는 문제점이 있다. 또 다른 방법으로는 span length를 이

용하는 방법이 있다[14]. 등고선 상의 한 점에서 다른 등고선 상의 한 점으로 연결된 선을 span이라 하고 그 span의 길이를 최소화하는 방법이다. 그 외에도 각각의 등고선의 중심으로부터의 점들의 방향을 연결하는 방법[9] 등 여러 가지가 있다. 그러나 이러한 방법들은 모두 등고선의 왜곡 현상이 심한 지형은 3차원 지형으로 표현하지 못한다는 문제가 있다.

마지막으로 지맥 문제는 하나의 등고선 안에 두 개 이상의 등고선이 존재하는 경우에 발생하는 문제이다. 하나의 등고선이 2개 이상의 등고선을 포함하면 내부의 등고선 사이에는 지맥(branch line)이 형성된다. 이 지맥을 어떻게 찾아서 표현할 것인가가 문제이다. 이 문제를 해결하기 위해서는 내부 등고선 사이에 가상의 등고선을 만들어야 하지만 어디에 점을 첨가해야 되는가는 매우 어려운 일이다. 따라서 복잡한 경우에는 지맥이 형성되는 곳의 지형을 나타내기 위한 점들을 사용자가 직접 결정해야할 경우도 있다[14].

### III. 지형의 3차원 시각화 기법

#### 3-1 3차원 시각화 과정

3차원 시각화를 위한 과정은 그림 2에서와 같이 크게 두 단계로 이루어진다. 첫 번째로 지형데이터를 읽어들이고 내부적인 데이터 구조인 등고선 트리로 구성하는 단계이다. 이 단계에서 등고선들에 대한 최소경계사각형(Minimum Bounded Rectangle, MBR)을 계산하여 등고선들간의 인접관계를 결정한다. 이 인접관계는 포함관계와 형제관계를 결정하므로 등고선 대응 문제를 해결하는데 사용된다.

그림 2. 시각화 과정  
Fig. 2. Visualization process.

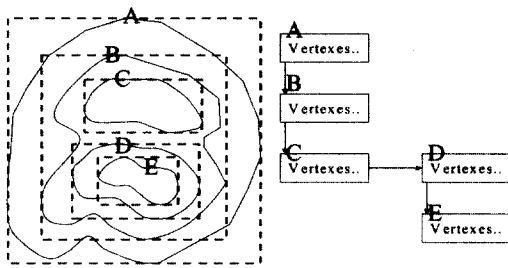


그림 3. 등고선간의 포함 관계  
Fig. 3. Inclusion relation between contour lines.

두 번째 단계는 등고선 트리를 탐색하면서 3차원 지형을 시각화하는 단계이다. 이 단계에서는 등고선 탐색을 통해 두 개의 등고선 사이의 면을 보간하는 기법을 사용한다. 특수한 지형의 경우 왜곡 현상방지를 위해 등고선의 확장기법을 통해 보간하며, 하나의 등고선만으로 이루어진 지형의 경우 가상의 중심점을 선정하여 보간 기법을 적용한다. 마지막으로 등고선 보간에 있어 하나의 등고선 내에 두 개 이상의 등고선이 존재하는 경우 NNS(Nearest Neighbor Search) 알고리즘을 이용하여 지맥 지점을 보간한다.

MBR을 이용하여 Correspondence를 문제 해결하기 위해 등고선은 완전한 폐곡선이라 가정하며 자료 구조는 높이와 포함관계를 고려하여 트리구조로 정의한다[7]. 각 등고선은 그림 3과 같이 등고선을 포함하는 MBR로 표현하고, MBR간의 포함관계를 이용하여 등고선 대응 문제를 해결한다.

### 3-2 등고선의 타일링

인접한 두 등고선 사이를 타일링하는 작업은 외부등고선과 내부등고선을 구성하는 점들을 이용하여 그림 4(a)와 같이 삼각형의 면들을 구성하는 것이다. 그림 4(a)의 두 등고선은 큰 왜곡현상이 없으므로 단순히 등고선상의 가까운 점들을 연결하면 자연스런 3차원 지형을 생성할 수 있다. 그림 4(b)는 두 개의 등고선을 삼각 면으로 타일링한 결과를 보인 것이다.

인접 등고선에 대한 면구성 방법을 설명하기로

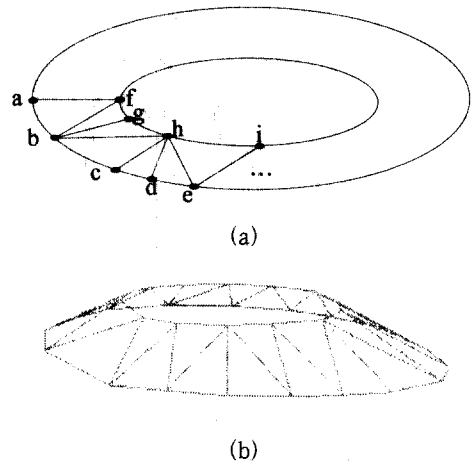


그림 4. (a) 일반적인 두 등고선의 관계, (b) 구현 예  
Fig. 4. (a) Typical relation between two contour lines, (b) Implementation example.

한다. 그림 4(a)에서 a를 외부등고선의 시작점, f를 내부등고선의 시작점으로 놓고 반시계방향으로 등고선들의 점들을 읽어 나갈 때, 다음과 같은 방법으로 각 점들을 연결하여 면을 구성할 수 있다. b를 외부등고선의 다음 점, g를 내부등고선의 다음점이라 놓으면, a를 중심으로 fg가 시계방향으로 진행하면 왜곡이 없는 정상적 진행이며, 외부등고선의 점 f를 중심으로는 ab가 반시계방향으로 진행하면 정상적 진행이다. 여기서 내부와 외부등고선의 기준점인 a와 f가 모두 왜곡이 없다면 두 점 사이의 거리  $|a-g|$ 와  $|b-f|$ 의 값 중에서 더 작은 거리의 점들을 연결한다. 만약 두 기준 점 중 어느 한쪽만 정상 진행이 가능하다면 그 점을 중심으로 삼각형 면을 구성한다. 그림에서는 bf의 거리가 더 작으므로 abf의 삼각 면이 구성되었다. 다시 점 b와 f를 중심으로 동일한 방법으로 왜곡검사를 실시하고 삼각 면을 결정해 간다.

## IV. 등고선 확장 알고리즘

3-2절의 방법에서 내부와 외부등고선의 기준점 모두에서 방향성이 만족되지 않으면 정상적인 삼각면의 구성이 불가능한 지형왜곡현상이 발생한 것이다. 본 절에서는 지형왜곡현상이 발생한 경우 등고

선의 확장을 통해 정확한 3차원 지형을 시각화할 수 있는 알고리즘을 제시하고자 한다. 지형왜곡의 해결 방안은 내부등고선의 왜곡에 의한 경우와 외부등고선의 왜곡에 의한 경우로 나누어 설명하기로 한다. 그리고 산의 정상과같이 등고선이 하나 뿐인 경우에 대한 시각화도 설명한다.

4-1 내부등고선의 왜곡

먼저 내부등고선의 심한 왜곡에 의해 정상적인 삼각 면 구성을 할 수 없는 경우에 내부등고선 내의 오목점을 검사하는 방법과 왜곡 현상을 보간하는 기법을 소개한다.

그림 5(a)의 등고선과 같이 등고선이 심하게 왜곡된 경우는 3-2절에서 설명한 알고리즘에 의해서는 bde와 같은 삼각 면이 생성된다. 이는 내부등고선의 다른 점 c를 포함하게 되어 잘못된 3차원 지형이 생성되는 것이다.

왜곡점의 발생은 점 b, d가 이루는 직선의 방정식에 c를 대입한 값의 부호와 점 b, e가 이루는 직선의 방정식에 c를 대입한 값의 부호가 달라지므로 탐지할 수 있다. 점 c가 왜곡점이 된다. 진행방향이 반시계방향일 경우에는 b의 앞에 있는 볼록점 a와 점 c를 연결하는 선분의 중심점에 가상의 점을 삽입하여 내부 등고선의 왜곡을 해결한다. 그림 5의 예에서 내부등고선의 확장점은 내부등고선의 점 중에서 현재점에서 가장 가까운 인덱스에 위치한 볼록점 a와 직선의 방정식에 의해 왜곡의 발생을 야기한 점 c의 중점이 된다. 따라서 두 점의 중점에 새로운 가상의 점을 생성시킨다. 그림 5(b)에서 보면 점 a와 점 c의 중점을 h라 하면, 내부등고선은 ahc로 확장된다. 이 때 중심점 h의 높이는 내부와 외부등고선의 중간 높이를 가진다. 그리고 내부등고선의 왜곡으로 폐쇄영역이 발생한 ac 안쪽의 공간은 단일 등고선 처리 방법을 이용하여 시각된다. 이 방법은 4-3절에서 설명하기로 한다.

전체적인 등고선 확장 알고리즘이 그림 6에 있다. 볼록점과 오목점의 탐색 방법은 [8]에서 상세히 설명하고 있다.

```

gen_3D(ic, ec)
//등고선 ic(내부등고선), ec(외부등고선)로 3차원 시각화
{SIP ← IP ← x 값이 가장 작은 ic 상의 점; // IP : 내부등고선의 기준점, SIP: 시각점
 SOP ← OP ← IP에서 가장 가까운 ec 상의 점; // OP : 외부등고선의 기준점

while (IP != SIP) // 시각점에 다시 도달하기까지 반복
{next_IP ← next_Point(IP, ic); next_OP ← next_Point(OP, ec);

if (direction(OP, next_OP)==CCW && direction(IP, next_IP)==CW)
{if ((|IP - next_OP| > |OP - next_IP|)
{gen_Triangle(IP, OP, next_IP); //삼각면 생성
IP ← next_IP; next_IP ← next_Point(IP, ic);
}
else
{if distortion(IP, next_OP)
process_Inside_Contour(IP, next_OP, ic); //내부등고선의 왜곡처리
else
{gen_Triangle(IP, OP, next_OP); //삼각면 생성
OP ← next_OP; next_OP ← next_Point(OP, ec);
}
}
else if (direction(OP, next_OP)==CCW)
{gen_Triangle(IP, OP, next_IP);
OP ← next_OP; next_OP ← next_Point(OP, ec);
}
else if (direction(IP, next_IP)==CW)
{if distortion(IP, next_OP)
process_Inside_Contour(IP, next_OP, ic); //내부등고선의 왜곡처리
else
{gen_Triangle(IP, OP, next_OP); //삼각면 생성
OP ← next_OP; next_OP ← next_Point(OP, ec);
}
}
else
{temp ← OP ← find_CCW(OP); //IP에 대해 direction(OP, next_OP)==CCW인 OP 탐색
process_Outside_Contour(temp, OP, ec) //외부등고선의 왜곡처리
}
} // while-end
    
```

그림 5. 내부등고선의 왜곡에 대한 확장

Fig. 5. An Extension to the distortion of the internal contour line.

그림 6. 등고선 확장 알고리즘

Fig. 6. Contour line extension algorithm.

4-2 외부등고선의 왜곡

4-2-1 내부등고선의 현재점에서 왜곡

이 절에서는 외부등고선의 왜곡에 의해 삼각면의 구성이 불가능하여 내부등고선을 확장하는 경우에 대해 설명한다. 예로서 그림 7(a)의 경우를 보면 내부등고선의 현재점인 a와 외부등고선의 현재점인 b에서 더 이상의 진행이 이루어지지 않으므로 점 a에서 점 d로의 내부등고선 확장이 필요하다. 그림 7(b)는 두 번의 등고선 확장이 필요한 예를 보인 것이다. 내부등고선의 확장을 위해 추가되는 가상점은 내부 등고선의 점의 높이와 외부등고선의 현재점이 갖는 높이의 절반인 높이 값을 갖는다. 두번째로 확장된 점은 다시 원래의 가상점 높이와 외부등고선 높이의 절반을 갖는다. 이렇게 함으로써 보다 더 자연스러운 지형을 생성할 수 있다.

그림 7(a)의 경우 내부등고선의 확장점을 구하기

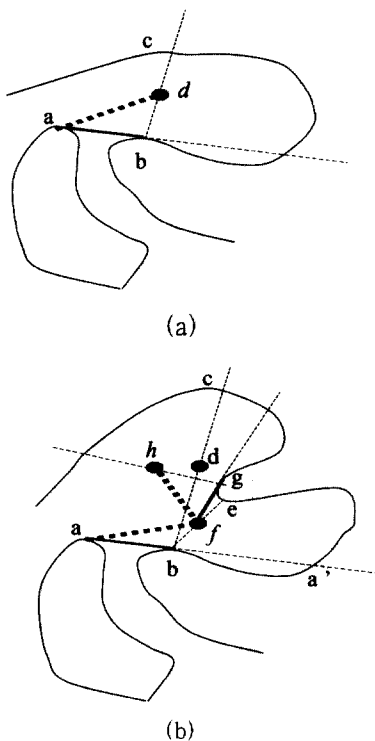


그림 7. 지형에 따른 내부등고선 확장 예 1  
 Fig. 7. Example 1: an extension of the internal contour line based on a terrain.

위해 우선 진행이 안되는 현재점인 a와 b를 연결한 선분 ab에 반시계방향으로 수직인 c를 선택한다. 따라서 가상의 등고선을 위한 확장점은 외부등고선의 점 b와 c의 중점인 d가 선택되고 내부등고선은 ...ada...와 같이 확장된다. 그러나 그림 7(b)와 같이 bc 사이에 또 다른 오목점이 존재하는 경우는 단순히 d와 같은 중심점이 확장점이 될 수 없다. 따라서 a'과 c사이에 있는 오목점을 탐색하고 이 중에서 |b-c|보다 거리가 더 가까운 오목점이 있으면 그 점 e를 왜곡된 지형의 특성을 나타내는 점으로 간주한다. 따라서 내부등고선의 확장점은 f가 되고 ...afa...로 내부등고선이 확장된다. 그림 7(b)의 경우는 확장 등고선을 따라 삼각면을 구성하다가 f와 g를 연결한 후 다시 왜곡현상이 발생하여 확장점 h로 내부등고선을 확장한다.

등고선의 확장점을 찾기 위해서는 내부등고선과 외부등고선의 현재점으로 구성되는 선분의 수직선과 외부등고선상의 오목점을 찾을 수 있는 방법이 필요하다. 가상점을 구하여 내부의 등고선을 확장시킬 경우에 필요한 점은 두 점의 좌표 값이 정해져 있고 그 점을 기준으로 수직인 점들을 외부등고선에서 찾는 것이며 외부등고선의 점들도 무작위로 저장되어 있는 것이 아니라 선을 만들어가며 저장되어 있기 때문에 그 두 가지 특성을 이용하면 가상점을 효율적으로 찾을 수가 있다. 자세한 알고리즘은 지면 관계상 생략하기로 한다[8].

4-2-2 내부등고선의 현재점이 아닌 곳에서 왜곡

위의 그림 7과 같이 외부등고선의 현재의 점이 왜곡을 일으키는 직접적인 원인이 되는 경우와 달리, 그림 8과 같이 외부등고선의 아직 읽어 들이지 못한 등고선에 의해서 왜곡이 일어나는 경우이다.

그림 8(a)에서 왜곡임을 알 수 있는 점은 외부의 a점과 내부의 d'점이다. 그 이유는 d' 점을 기준으로 외부등고선의 진행방향이 시계방향으로 변경되기 때문이다. 그러나 실제로 왜곡은 dp부터 발생하였으며 원인은 외부등고선의 점 b이다. 따라서 a와 가장 가까운 내부등고선의 점 c'까지 되돌아와 (c, c')를 시작점으로 하여 외부등고선의 현재점이 a가 될 때

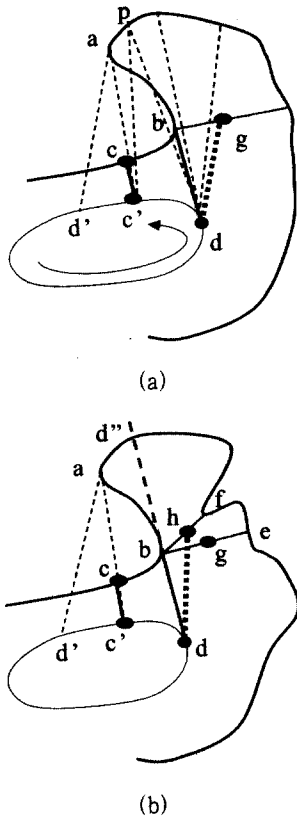


그림 8. 지형에 따른 내부등고선 확장 예 2  
 Fig. 8. Example 2: an extension of the internal contour line based on a terrain.

까지 반대방향으로 등고선을 읽는다. 이 과정이 끝나면 다시 (c, c')를 시작으로 외부등고선의 현재점이 제일 먼저 읽기를 시작한 점에 이르기까지 계속한다. 이 과정에서 외부등고선의 점 b와 내부등고선의 점d에서 왜곡이 있으므로(b점은 실제로 왜곡을 일으키는 점이므로) 선분 bd에 시계방향으로 수직인 점 e를 선택하고 외부등고선의 점 b와 e사이에 오목점이 없으므로 중심 점 g가 내부등고선의 확장점이 된다. 그림 8(b)는 중심점인 g대신 외부등고선의 ed"사이에 오목점 f가 존재하여 이를 연결한 선분의 중심점인 h를 확장점으로 선택한 경우이다.

4-3 단일 등고선인 경우

마지막으로 내부등고선의 왜곡으로 인해 발생한

단일 등고선이나 산 정상과 같은 단일 등고선도 면으로 시각화하여야 한다. 하나의 등고선을 면으로 시각화 하기 위해서는 등고선 내부에 새로운 내부등고선을 가상으로 생성시켜야 한다. 이러한 상황은 가장 높이가 높아 트리의 리프노드에 있는 등고선과 외부등고선과 연결되지 못하는 내부등고선에서 발생할 수 있다. 내부의 등고선을 가상으로 생성시키기 위해 먼저 등고선의 중점을 잡아야 한다. 등고선이 이루는 MBR의 중점이 반드시 등고선의 내부라 할 수 없으므로 본 논문에서는 MBR의 X축으로의 중점을 지나는 점들의 중점으로 Y값을 정하고 정해진 Y값을 갖는 점들을 이용하여 X의 중점을 구하여 전체 등고선의 중점을 정한다.

그림 9에서와 같이 먼저 X축의 중점을 지나는 등고선의 점들을 구한다. 그 후에 구한 점들의 중심점과 같은 Y값을 갖는 점들의 중점을 구한다. 이때 처음 X축의 중점은 점들의 쌍을 생각하지 않아도 무방하지만 같은 Y값을 갖는 점을 구할 때는 점들의 쌍을 고려해야만 등고선의 내부의 점을 구할 수 있다. 그 후에 구한 등고선 내부의 점을 기준으로 등고선을 외부등고선이라 가정한다. 내부의 등고선의 높이는 두 가지 경우에 따라서 다르게 설정되어지는데 먼저 트리의 리프노드에 위치하여 가장 높은 곳의 등고선을 나타내는 등고선인 경우에는 그 부모노드의 등고선의 높이 차를 계산하여 그 절반의 높이를 더한 값으로 한다. 또한 외부 등고선과 연결되어지지 못해 그 자체로 시각화되어야 하는 등고선은 외부등고선과의 높이 차보다 절반의 높이를 빼준 값으로 설정되어진다.

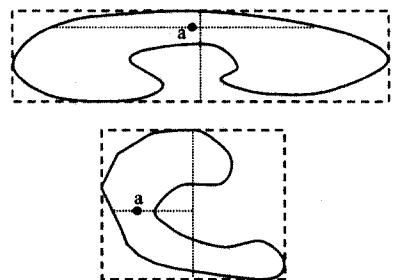


그림 9. 단일 등고선의 중점  
 Fig. 9. Center point on a single contour line.

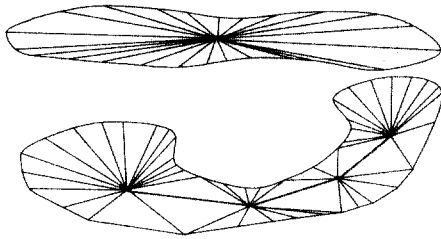


그림 10. 단일 등고선의 시각화  
Fig. 10. Visualization of a single contour line.

그림 10 은 하나의 등고선에 면을 구성한 예이다.

4-4 NNS 알고리즘을 이용한 지맥문제 해결

하나의 등고선은 같은 높이를 가진 여러 개의 등고선을 포함할 수 있다. 이러한 특성 때문에 발생 한 문제가 지맥문제이다. 외부의 등고선 하나와 여러 개의 내부등고선들 사이에서의 시각화를 위해서도 4-1절에서 다른 방법으로 왜곡의 생성 여부를 판단 한다. 단, 내부에 여러 개의 등고선이 있을 경우는 내부등고선의 현재의 점을 다른 등고선으로 옮겨야 한다. 그림 11(a)에서 등고선 A가 등고선 B, C를 포함하고 있으며, B부터 내부등고선과 외부등고선의 관계를 검사하면서 내부등고선의 현재점과 외부 등고선의 현재점 사이의 거리보다 더 가까운 점을 가진 다른 등고선이 검색되면 내부등고선의 현재점을 검색된 등고선으로 옮겨 주어야 한다.

A등고선의 b와 e의 거리보다 더 짧은 거리를 갖는 C등고선의 i점을 발견하고 내부등고선의 현재점을 C등고선의 i점으로 이동한다. 이때 두 등고선들 사이의 가상등고선은 점 b에서 연결할 수 있는 각 등고선의 점들의 중점, 즉, B의 f와, C의 h의 중점으로 하며, 가상등고선을 이루는 점 k의 높이값은 외부등고선과 내부등고선의 중간값으로 한다. 이때 k가 외부등고선의 지형 특성을 제대로 반영한 것으로 보기는 어렵기 때문에 선분 hk를 지나는 외부등고선이 검색되면 k는 무시하고 새로운 가상의 점 k를 결정한다. 또한 때면 외부등고선과 내부등고선을 연결할 때마다 다른 내부의 등고선에 더 짧은 거리로써 연결할 수 있는 점이 있는지를 검사하는 오버헤

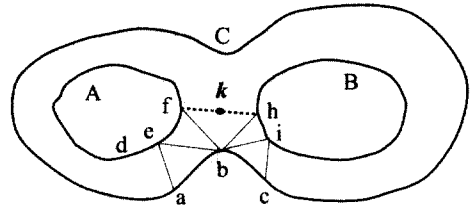


그림 11. 지맥 문제의 해결  
Fig. 11. A Solution to the branching problem.

드를 줄이기 위해 Nearest Neighbor Search 알고리즘을 응용한다[10]. 즉, 다른 외부등고선 MBR의 MINDIST와 MINMAXDIST를 이용하여 현재의 내부등고선의 점과 외부등고선의 점이 이루는 길이가 외부등고선의 현재점과 다른 등고선의 MINDIST와 MINMAXDIST사이에 있을 때 검사함으로써 오버헤드를 줄일 수 있다.

4-5 구현 예

앞의 4절에서 외부등고선과 내부등고선의 관계를 고려하여 내부등고선을 확장함으로써 내부등고선과 외부등고선을 닮은꼴의 등고선으로 확장하였으며, 시각화 과정에서는 닮은꼴로 이루어진 등고선을 따라가며 삼각형으로써 면을 완성한다. 시각화 시에도 물론 내부등고선과 외부등고선의 점들이 이루는 방향이 시계방향인지 반시계 방향인지를 파악하고 그 후에 거리로써 면을 이룰 점을 결정하여준다. 이때 가상의 등고선을 만들어 주었기 때문에 단지 내부등고선의 현재점과 다음점, 그리고 외부등고선의 현재점과 다음점의 시계방향과 반시계방향만으로 탐색방향을 결정하게 되면 가상등고선이 만들어지는 점에서 내부등고선의 이동이 멈추어버리는 상황이 발생할 수 있다. 따라서 가상등고선까지 완성된 후의 시각화를 위해서는 내부등고선의 보다 진행방향으로 두 개의 점까지도 고려를 해서 시각화해야 완전한 시각화를 이룰 수 있다.

그림 12와 그림 13은 실제 구현한 프로그램을 기반으로 입력데이터를 등고선의 확장과정을 거친 후 시각화한 모양이다. 그림 12의 (a)와 그림 13의 (a)는 각각 입력 데이터가 되는 등고선의 2차원 표현이



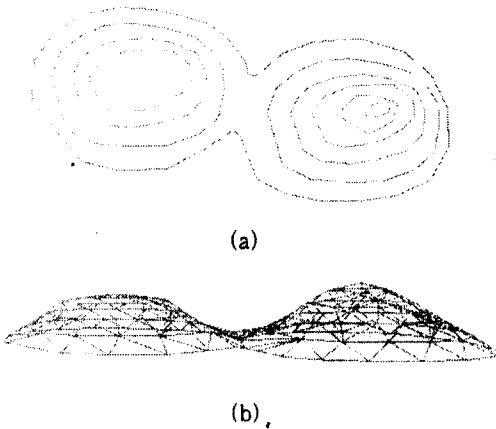


그림 12. 단순 지형에 대한 시각화의 예  
Fig. 12. An example of visualizing a simple terrain.

며 그림 12의 (b)는 그림 13의 (b)는 시각화한 등고선을 각각 180°도 시점과 45°도 시점으로 출력한 예다. 그리고 각각의 구형 예는 3차원으로 디스플레이된 결과이다.

본 논문에서 제시한 시각화 알고리즘은 Windows 98을 운영체제하에서 Visual C++로 구현되었으며

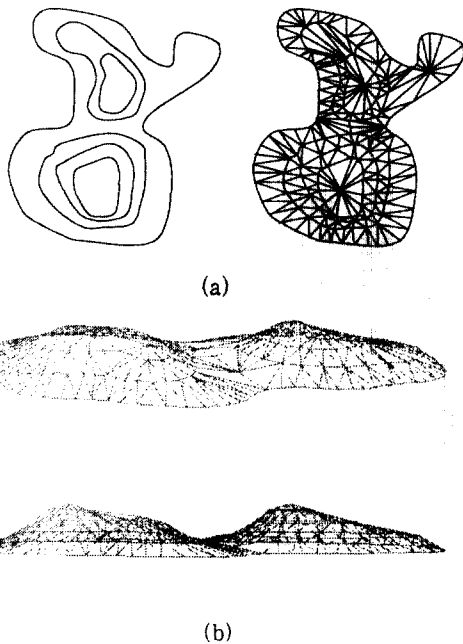


그림 13. 복잡한 지형에 대한 시각화의 예  
Fig. 13. An example of visualizing a complex terrain.

그래픽 API는 OpenGL을 사용하여 구현하였다[15], [16].

### V. 결 론

본 논문에서는 CAD에서 주로 사용되는 화일 포맷인 DXF화일에서 직접 지형데이터를 추출하여 지형을 3차원으로 시각화할 수 있는 기법을 제시하였다. 지금까지 지형의 시각화를 위한 데이터는 인공 위성 영상을 이용하거나, 이미지 프로세싱을 통해 지도에서 필요한 데이터를 직접 획득하는 방법이 대부분이다. 그러나 이러한 방법은 전처리 과정이 복잡하고 3차원 지형의 생성에 필요한 데이터가 방대하다는 단점이 있다. 본 논문에서 제시한 방법은 지형의 3차원 표현을 위해 필요한 데이터의 추출과 보간 작업 등을 자동화하고 왜곡된 등고선을 닳은꼴의 등고선으로 확장함으로써 정확하고 세밀한 3차원 지형의 묘사를 가능토록 하였다.

추후에는 2차원 수치지도 형식인 DXF 화일을 직접 입력데이터로 사용할 수 있는 방안을 연구할 예정이다. 이를 위해서는 지리적으로 이어진 여러 개의 DXF화일을 연결하여 시각화하는 방법이 필요할 것이다. 또한 항공기 시뮬레이터나 자동차 항법시스템과 같이 실시간으로 지형의 변화가 요구되는 응용에 적용하기 위해 지형의 해상도를 조절할 수 있는 방안도 필요할 것이다.

### 참 고 문 헌

- [1] R. Kasturi, et al, "Map data processing in geographical information systems", *IEEE Computer*, vol. 22, no. 12, pp. 10-21, December 1989.
- [2] 지승도, 황수찬, 채수환, "차세대형 교통정보시스템", 한국항공대학교 전자정보통신공학논문지, 1998. 10.
- [3] 이진선, 정성종, "등고선 지도로부터 3차원 지형의 복원을 위한 래스터기반 알고리즘", 정보과학회 논문지, 제 22권, 제8호, 1995. 8.
- [4] 황수찬, 지승도 외, "산불확산의 효율적 방재

를 위한 실시간 모니터링 및 예측시스템 개발에 관한 연구”, 최종 연구개발 결과 보고서, 한국항공대학교, 1998.

- [5] B. Makarovic, “Progressive Sampling for Digital Terrain Models”, *ITCJ.*, no. 1973-3, pp. 397-416, 1973.
- [6] L. L. Scarlatos and T. Pavlidis, “Adaptive Hierarchical Triangulation”, *auto-Carto 10, Technical Papers, ACSM-ASPRS Annual Convention*, vol. 6, pp. 234-246, 1991.
- [7] 김수선, “등고선 데이터로부터 3차원 지형복원”, 박사학위논문, 1997.
- [8] 박재민, 황수찬, “3차원 지형의 시각화를 위한 등고선 확장기법”, *Korean Database Conference 2000 논문지*, 한국항공대학교, 2000.
- [9] L. T. Cook, P. N. Cook, K. R. Lee, S. Batnitzky, B. Y. S. Wong, S. L. Fritz, J. Ophir, S. J. Dwyer, L. R. Bigongiari, and A. W. Templeton, “An algorithm for volume estimation based on polyhedral approximation”, *IEEE Trans. Biomed. Eng. BME-27*, Sept. 1980, 493-500.
- [10] N. Roussopoulos, S. Kelley, and F. Vincent.

“Neareat neighbor queries”, In *Proceedings of the ACM SIGMOD International Conference on the Management of Data*, pages 71-79, June 1995.

- [11] D. Meyers and S. Skinner, “Surface from Contours”, *ACM Transactions on Graphics*, vol. 11, no. 3, July 1992, Pages 228-258.
- [12] E. Keppel, “Approximating complex surfaces by triangulation of contour lines”, *IBM J. Res. Dev.* Jan. 1975, 2-11.
- [13] H. Fuchs, Z. M. Kedem, and S. P. Uselton, “Optimal surface reconstruction from planar contours”, *Commun. ACM* 20, Oct. 1977, 693-702.
- [14] H. N. Christiansen, T. W. Sederrerg, “Conversion of complex contour line definitions into polygonal element mosaics”, *Compt. Graph.* 12, Aug. 1978, 187-192.
- [15] K. Rule, “*3D Graphics File Formats: A Programmer's Reference*”, Addison-Wesley, 1996
- [16] 신영수, 김현석, “3차원 그래픽-C언어로 배우는 이론과 알고리즘”, 가남사, 1995.

## 한 정 규(韓正奎)

1995년 : 한국항공대학교 전자계산학과(이학사)  
 1997년 : 한국항공대학교 컴퓨터공학과(공학석사)  
 1997년 3월~현재 : 한국항공대학교 컴퓨터공학과 박사과정  
 관심분야 : 데이터베이스 시스템,

웹 마이닝, GIS

## 백 중 환(白重煥)

1981년 2월 : 한국항공대학교 항공통신공학과(공학사)  
 1987년 7월 : 미국 오클라호마주립대 대학원 전자공학과(공학석사)  
 1991년 7월 : 미국 오클라호마주립대 대학원 전자공학과(공학박사)  
 1981년 6월~1984년 10월 : (주)

동아엔지니어링 근무

1991년 9월~1992년 2월 : 한국전자통신연구소 선임연구원  
 1992년 3월~현재 : 한국항공대학교 항공통신정보공학과 부교수

관심분야 : 컴퓨터비전, 영상처리, 영상압축 및 항공기 시뮬레이터 개발 등임.

**황 수 찬(黃壽贊)**

1984년 2월 : 서울대학교 전자계  
산기공학과(공학사)

1986년 2월 : 서울대학교 대학원  
컴퓨터공학과(공학석사)

1991년 2월 : 서울대학교 대학원  
컴퓨터공학과 (공학박사)

1995년~1996년 : 미국 캘리포니

아주립대 방문교수

1991년 3월~현재 : 한국항공대학교 컴퓨터공학과 부교수

관심분야 : 멀티미디어 데이터베이스, 객체지향 데이터  
베이스, 지리정보시스템 등임.