

특이점 추출을 통한 지형데이터의 빠른 삼각망 생성

전경훈 · 구자영

단국대학교 정보컴퓨터 학부

Fast Triangulation of Terrain Data through Unique Point Extraction

Kyong-Hun Jeon and Ja-Young Koo

Division of Computer Science and Statistics Dankook University

Abstract : Triangulated irregular network is one of the most frequently used terrain modeling methods. It represents terrain precisely using only a small amount of data, and enables fast rendering of terrain. In this paper, ridges and valleys are extracted from the terrain, and used as a set of vertices for the construction of triangulated irregular network. An experiment shows the new method reduces the time for construction of the triangulated irregular network significantly with almost equal amount of induced error.

Key Words : Triangulated Irregular network(TIN), Delaunay Triangulation, Gridy Insertion, Ridge Detection.

요약 : 불규칙 삼각망은 대표적으로 사용되는 지형 모델링 방법 가운데 하나이다. 이 방법은 적은 데이터 량으로 지형의 특징을 잘 표현할 수 있고, 렌더링 시간을 단축시킬 수 있다. 본 논문에서는 능선 검출 알고리즘을 이용하여 지형데이터로부터 능선과 계곡을 검출하고, 이를 불규칙 삼각망의 구성을 위한 정점들의 집합으로 사용함으로써 기존 방식과 거의 동등한 오차수준에서 삼각망의 구성시간을 현저하게 단축시키는 방법을 제안하고 있다.

1. 서 론

지형데이터는 두 개의 변수로 이루어진 함수로 규정되는 고도필드(height field)의 한 사례로서, 비행시뮬레이션, 비디오게임을 비롯한 다양한 응용들에서 사용된다. 지형데이터의 용량은 일반적으로 상당히 크기 때문에, 빠른 디스플레이를 위한 효율적인 자료구조의 선택은 핵심적 문제가 된

다. 대부분의 그래픽스 시스템의 하드웨어가 다각형의 고속렌더링을 지원하므로 지형데이터를 다각형 메쉬(polygonal mesh)형태로 표현하는 것은 타당성을 갖는다. 원시데이터는 보통 사각격자(rectangular grid)의 형식을 갖는데, 이웃하는 정점들을 변으로 연결함으로써 손쉽게 다각형 메쉬로 변환될 수 있다. 그러나 이 경우 렌더링 되어야 할 다각형의 수가 너무 많아서 비효율적이 된다. 예

를 들어, 10001000 격자 데이터의 경우 2,000,000개의 삼각형으로 표현되고 이것은 실시간 렌더링에 적합하지 않은 데이터 량이다.

사각격자 표현의 이러한 문제점을 극복하는 방식의 하나인 불규칙 삼각망(triangulated irregular network, TIN)은 지형데이터의 부분들을 삼각형 조각으로 근사하고, 전체 지형데이터를 연결된 삼각형 조각들로 표현하는 것이다. 이렇게 함으로써 주어진 오차의 한계 내에서 정점의 수를 줄일 수 있고 빠른 렌더링이 가능해진다. 본 논문에서는 Delaunay 삼각망(Guibas and Stolfi, 1985; Lawson, 1977)을 다룬다. 이것은 입력 점들의 투영 공간상에서 삼각망을 구성하는 방법으로서, 삼각형들의 최소 내각들을 최대화하는 특성을 가지며 이를 통해서 가늘고 긴 부자연스러운 삼각형의 가능성을 줄여준다. 구현방법으로는 상세화(refinement)가 주로 사용되는데, 초기의 개략적 삼각망에서 출발하여 허용오차가 만족될 때까지 매 사이클마다 새로운 정점을 삼각망에 더하는 과정을 반복함으로써 상세한 삼각망을 구성해 간다.

흔히 사용되는 삽입방법은 그리디 삽입 알고리즘(greedy insertion algorithm)으로(Floriani *et al.* 1983), 하나의 정점을 삽입하기 위해서 아직 삽입되지 않은 모든 정점들에 대한 각각의 지역예러를 구하고 최대오차를 갖는 정점을 찾는다. 이러한 재계산과 재비교 연산이 속도를 저하시키는 요인이 된다. 이러한 문제를 해결하기 위한 방법으로 점진적인 삼각망 생성과정의 변화가 지역적이라는 사실을 이용하여 중복된 계산을 줄이는 방법이 사용되기도 하였고(Garland and Heckbert, 1995), 입력된 지역정보를 구역화하여 각각 삼각망을 구성한 후 통합하는 방법이 사용되기도 하였다(강윤식 *et al.*, 2000). 그러나 이러한 방법들이 탐색영역을 줄이는 효과는 있으나 정해진 탐색영역 내의 모든 점들에 대한 계산이란 점에서 공통점을 갖는다.

본 논문에서는 최대오차를 유발하는 점들이 대부분의 경우 지역적으로 극대값 혹은 극소값을 갖는 특이점들이라는 점에 유의하여 이러한 특이점

들을 먼저 추출한 후 이 점들을 삼각망의 구성을 위한 정점들의 집합으로 사용하는 방법을 제안하고 있다. 전체 점들에 비해서 특이점들은 밀도가 낮으므로 계산량을 줄일 수 있음을 보였다. 예를 들어, 앞에 언급된 지형데이터의 분할과 삼각망의 통합방식의 경우, 지형데이터가 분할됨으로써 삼각망과 지형의 최대 오차 유발 지점을 찾는 시간이 줄어들지만, 본 논문에서 제안된 방법을 적용하는 경우 줄어든 지역 내에서도 특이점에 대해서만 오차계산이 일어남으로써 더욱 빠르게 모델이 구성될 수 있다.

2절에서는 Delaunay 삼각망과 그리디 삽입 알고리즘을 소개하고, 3절에서는 특이점 추출을 이용한 삼각망 생성을 제안하며, 4절에서는 실험결과를 보이고, 5절에서는 결론을 기술한다.

2. Delaunay 삼각망과 그리디 삽입 알고리즘

고도필드 $H(x, y)$ 로부터 샘플링 된, 각기 $q_i = (x_i, y_i, h_i)$ 로 표현되는 3차원 정점들의 집합 $Q = \{q_1, \dots, q_n\}$ 가 주어질 때, 문제는 어떻게 인접하는 세 개의 정점들로 그루핑 함으로써 3차원 공간상에서 연결된 삼각형들의 집합으로 $H(x, y)$ 를 근사할 수 있는가 이다. 이 문제의 해결책으로 Delaunay 삼각망의 구성을 통한 근사화 방식들이 제안되었다(Guibas and Stolfi, 1985; Lawson, 1977). $p_i = (x_i, y_i)$ 를 q_i 의 xy 평면상에 투영된 점으로서 사이트(site)라고 부를 때, Delaunay 삼각망은 사이트들의 집합 $P = \{p_1, \dots, p_n\}$ 에 대해서 적용된다. Delaunay 삼각망은 Voronoi 다이어그램의 쌍대(dual)이다. Voronoi 다이어그램은 Euclidean 평면을 각 사이트 p_i 에 가장 가까운 점들의 집합인 Voronoi 영역 $V(p_i)$ 로 분할한 것이다(Fig. 1.(a)):

$$V(p_i) = \{x \mid |p_i - x| \leq |p_j - x|, \forall j \neq i\}.$$

Delaunay 삼각망은 Voronoi 다이어그램에서 인접하는 Voronoi 영역에 해당하는 사이트를 간선으로

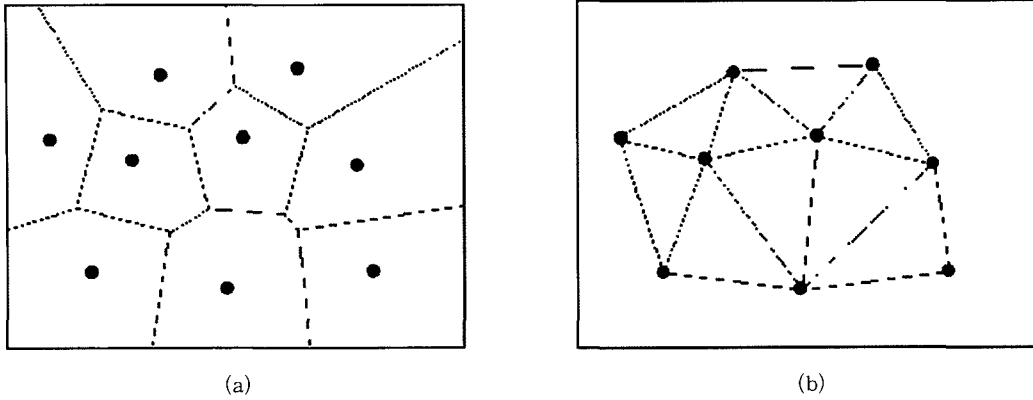


Fig. 1. (a)Voronoi diagram (b)Delaunay triangulation.

연결한 그래프이다(Fig. 1.(b)).

Delaunay 삼각망의 구현방법으로 그리디 삽입 알고리즘이 사용된다(Floriani *et al.*, 1983). 지형의 모서리에 해당하는 네 개의 정점으로 초기 삼각망을 생성한다. 이 삼각망은 두 개의 삼각형을 포함하고, 전체 지형을 개략적으로 근사한다. 이 초기 삼각망에 지형의 특징을 가장 잘 표현할 수 있는 특징점을 반복적으로 선택, 삽입시켜 원하는 정밀도를 갖는 삼각망을 생성한다. 특징점을 선택하기 위한 방법으로 지역오차가 사용된다. 지역오차 $e(p)$ 는 지형상의 특징점 $p=(x, y, h)$ 의 h 값과, p 와 같은 x, y 값을 갖는 삼각형상의 점 $p'=(x, y, h')$ 의 h' 값과의 차이이다. 새로운 정점이 삽입되었으므로 삼각망을 재구성한다. 삼각망의 재구성은 Fig. 2에 보이는 바와 같다(Garland and Heckbert, 1995). 기존의 삼각형 ZBD 내부에 정점 A가 삽입

되면(Fig. 2(a)), 정점 A와 Z, B, D 를 각각 연결하여 세 개의 간선을 새로 만든다. 기존의 간선 BD, DZ, ZB가 재구성의 대상이 되는데 BD의 재구성 여부를 판단하기 위해서 삼각형 BCD의 외접원을 그려서(Fig. 2(b)) 새로이 추가된 정점 A가 외접원 밖에 있으면 재구성되지 않고, 외접원 안에 있으면 간선 BD를 간선 AC로 치환한다(Fig. 2(c)). 이렇게 함으로써 Delaunay 삼각망의 조건을 계속 유지하게 된다.

3. 특이점 추출을 이용한 삼각망 생성

1) 능선검출(Ridge Extraction)에 의한 특이점 추출

본 논문에서 사용한 능선검출(ridge extraction)

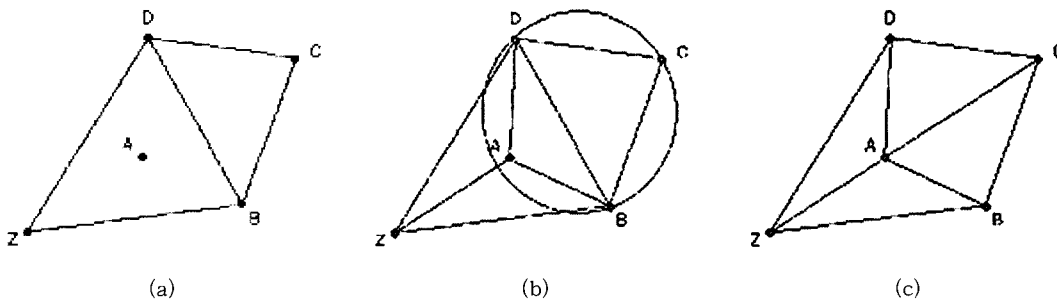


Fig. 2. Vertex insertion and modification of Delaunay triangulation.

방식은 고도필드의 역수에서 계곡을 찾는 것과 같으며 빗방울이 모여 흐르는 경로를 찾는 방법과 동일하다(Guo and Richardson, 1998). 먼저 방향성 그래프 $G(V, E)$ 를 구성하는데, 여기서 정점의 집합 V 는 대상 영역의 모든 점들이고 간선의 집합 E 는 다음의 관계를 만족하는 정점의 쌍이다:

$$E = \{ \langle u, v \rangle \mid \forall u, v \in V, v.oDir = u \}.$$

여기서 $v.oDir$ 는 v 의 이웃 정점들 중 최대상승방향에 있는 정점을 나타낸다. 그래프 G 에서 둘 이상의 정점들로부터 최대상승방향에 위치하는 정점들의 집합을 *SMP*(Stream Merge Points)라고 부르며 능선에 해당하는 점들의 집합이 된다. 능선을 구하는 알고리즘은 다음과 같다. 여기서 $v.oDefined$ 는 정점 v 에서 다른 정점으로 향하는 간선이 정의되어 있는지를 나타내는 플래그이고, $u.streamIn$ 은 정점 u 로 향하는 간선의 수를 나타낸다. 먼저, 모든 점에서 가장 급격한 기울기로 상승하는 이웃으로 간선을 달아 줌으로써 그래프를 완성하는데 이 그래프에서 하나 이상의 이웃으로부터 간선이 연결된 점이 능선점이 된다. 역으로 생각하면 하나 이상의 이웃으로부터 빗물이 흘러올 수 있는 점이 계곡점이 되는 것과 마찬가지이다. 이 때 데이터에는 노이즈가 포함될 수 있는데 이러한 작은 변화가 모두 능선으로 검출되는 것을 방지하기 위해서 전처리 과정으로 저역필터링(low-pass filtering)

을 적용한다. 본 논문의 실험에서는 3×3 의 Gaussian low-pass filter를 2회 적용했다.

```
Function RIDGE( $H(x, y)$ )
Clear SMP
Construct graph  $G(V, E)$  for  $H(x, y)$ 
for each ( $v \in V$ )
    if ( $v.oDefined$ )
         $u \leftarrow v.oDir$ 
         $u.streamIn \leftarrow u.streamIn + 1$ 
for each ( $v \in V$ )
    if ( $u.streamIn > 1$ )
         $SMP \leftarrow SMP \cup \{v\}$ 
return SMP
```

Fig. 3은 위의 알고리즘의 적용 예를 보인다.

2) 특이점을 이용한 삼각망 생성

그리디 삽입방식에 의해서 삼각망을 생성하는 상세화(refinement)방법은 대상지역의 네 구석을 이용해서 두 개의 삼각형으로 구성된 초기삼각망으로부터 시작된다. 매 반복마다 현재의 삼각망과 지형데이터의 차이가 가장 큰 점을 찾아내어서 기존의 삼각망에 그 점을 삽입하고 삼각망을 재구성하는 것이다. 이때 최대오차를 유발하는 정점을 찾기 위해서 많은 비교연산이 일어나게 된다. 물

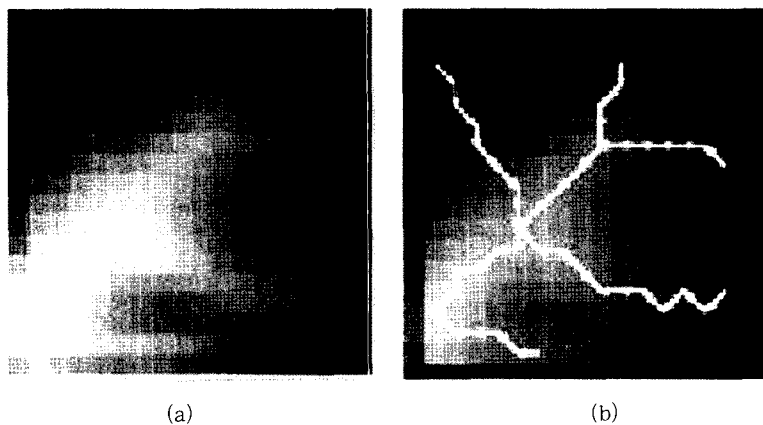


Fig. 3. (a) Height field (b) Ridge points extracted from (a).

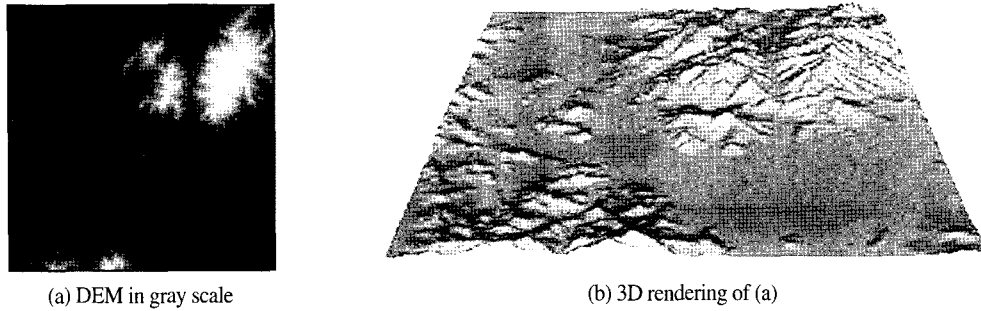


Fig. 4. 200×200 DEM used in the experiment.

른 점진적인 삼각망 생성과정의 변화가 지역적이라는 사실을 이용하여 중복된 계산을 줄이는 방법이 사용되기도 하였고(Garland and Heckbert, 1995), 입력된 지역정보를 구역화하여 각각 삼각망을 구성한 후 통합하는 방법이 사용되는 등(강윤식 *et al.*, 2000), 비교연산을 줄이기 위하여 탐색영역을 줄이는 다양한 방법이 제안되었지만, 줄여진 탐색영역 내에서는 모든 점들에 대해서 비교연산을 시행하여야한다. 본 논문에서는 3.1절에 기술한 능선과 계곡을 이루는 특이점들을 삼입대상의 정점집합으로 한정함으로써 비교연산의 수를 대폭 줄일 수 있었다. 다음은 특이점 추출을 이용한 지형 근사화에 대한 의사 코드이다. 먼저 3.1절에 기술한대로 Low-pass filtering을 함으로써 작은 지형적 굴곡에 의해서 과도한 능선이 검출되는 것을 방지한다. 주어진 데이터에 대해서 RIDGE 함수를

적용하여 능선을 추출하고, 데이터의 역수에 대해서 다시 한 번 RIDGE 함수를 적용함으로써 계곡을 추출하여 그 둘을 합하여 특이점으로 사용한다. 초기에 대상 지역의 네 개의 코너에 있는 자료 점 중에서, 대각 위치의 두 개를 연결하면 두 개의 삼각형이 형성되고 이것이 초기의 삼각망으로 사용된다. 특이점 중에서 현재의 삼각망과 가장 큰 오차를 유발하는 특이점을 탐색해서 삼각망에 삼입하고, 2절의 방식에 따라 Delaunay 조건을 만족하도록 삼각망을 재구성한다. 이 과정을 주어진 오차의 한계에 이르기까지 반복함으로써 삼각망을 완성한다.

Lowpass filtering $H(x, y)$ // 소규모의 능선을 제거하기 위한 smoothing
 $P = \text{RIDGE}(H(x, y))$ // 능선추출

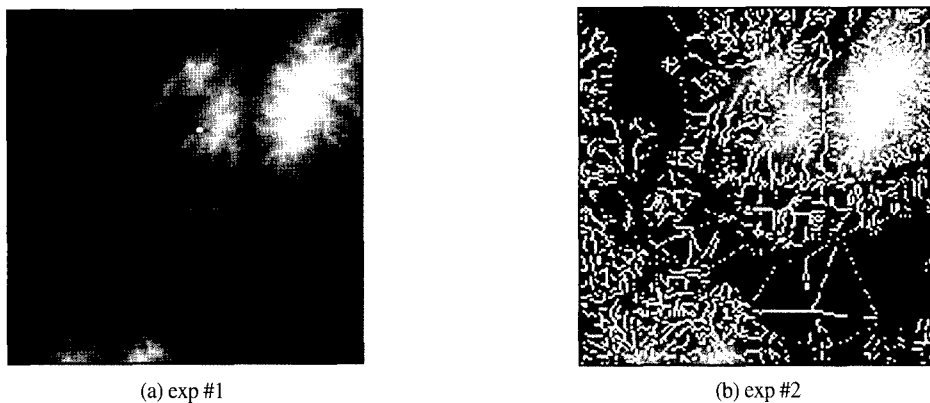


Fig. 5. Set of candidate vertices for refinement. (a) all the points (b)white points.

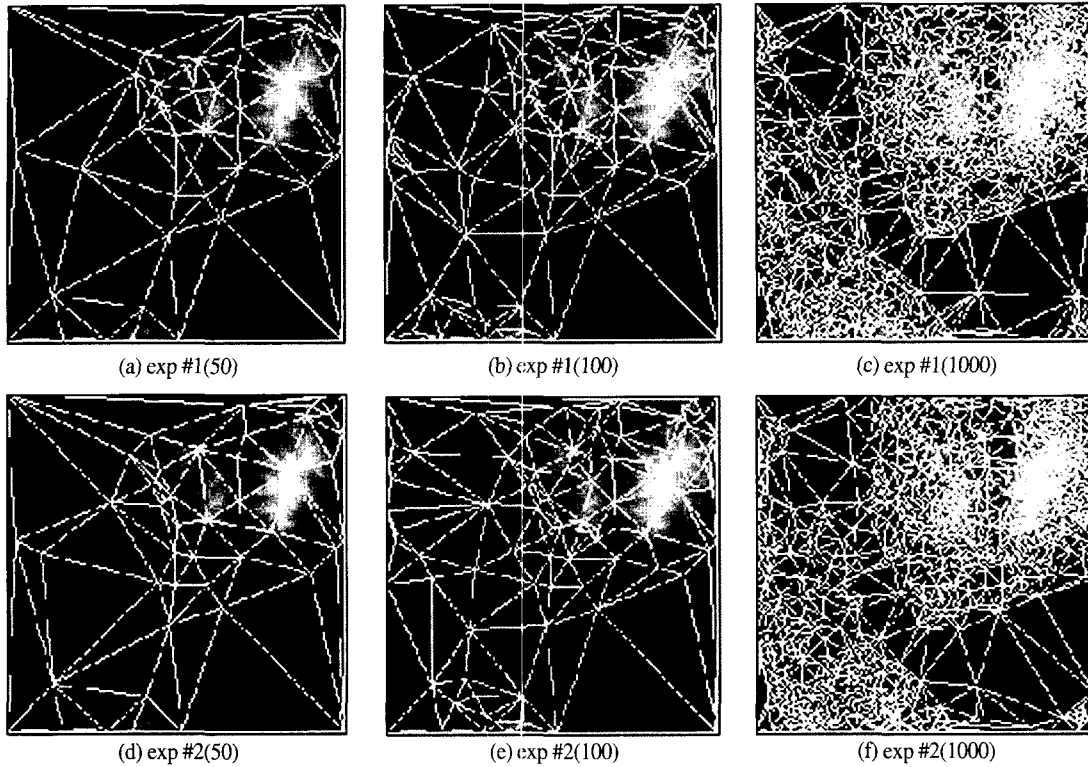


Fig. 6. Result of two algorithms(number of vertices).

```

P = P ∪ RIDGE(-H(x, y)) // 계곡추출
Initial triangulation // 대상지역의 네 구석을 이용 두 개
의 삼각형으로 삼각망구성
while not Goal_Met() do // 정해진 회수 혹은 오차조건이
만족할 때까지 반복
    search p ∈ P that maximize the local error e(p)
    // 최대오차의 특이점 탐색
    insert p into the triangulation
    // 기존의 삼각망에 새로운 정점을 삽입
    modify the triangulation
    // Delaunay 조건을 만족하기 위한 삼각망 재구성
    delete p from P
end while
    
```

4. 실험결과

실험에 사용된 지형 정보는 Fig. 4와 같이 의 크기를 갖는 DEM 자료이며, 산악 지형과 평지가

적절히 분포되어 있다.

실험 1은 입력된 지형의 모든 정점을 대상으로 상세화를 수행하고, 실험 2는 추출된 능선과 계곡 점들을 대상으로 상세화를 수행하였다. 상세화 대상으로 사용되는 정점의 수는 실험 1의 경우 39,996개이고 실험 2의 경우 7,283개이다(Fig. 5).

Fig. 6은 위의 방법대로 실험한 후, 각각 50개, 100개, 1000개의 특징점을 추출하고, 이를 Delaunay 삼각망으로 나타낸 것이다. 예를 들어, exp #1(50)은 주어진 지형상의 모든 점을 대상으로 하여 상위 50 개의 점으로 삼각망을 구성한 결과이고, exp #2(100)은 추출된 특이점들(능선 및 계곡점)을 대상으로 상위 100 개의 점으로 삼각망을 구성한 결과이다.

삼각망과 지형 데이터 사이의 평균 오차를 지형데이터의 최대 고도차이로 나눈 백분율을 삼각망의 평가지표(E)로 사용한다:

$$E = \frac{1}{N} \cdot \sum_{i=1}^N \frac{\Delta h_i}{\Delta H} \times 100.$$

여기서 N 은 입력된 지형의 정점 수이고, $\Delta h_i = |h_i - h_i'|$ 는 i 번째 정점에서의 오차이고, $\Delta H = |H_{max} - H_{min}|$ 는 주어진 지형데이터의 최대 고도차이다.

Table 1과 Fig. 7은 여러 정점 추출 개수에서 실험 방법별 평균 오차율과 지역 에러 계산 회수를 나타낸다. 본 논문에서 제안한 삼각망 구성 방식이 거의 동등한 오차 수준에서 상당한 계산량의 감소를 가져옴을 볼 수 있었다. 예를 들어, 1000개의 특이점으로 삼각망을 구성하는 경우 오차율에서는 1.08%에서 1.09%로 0.01%포인트의 증가가 일어난 반면, 계산량에 있어서는 718,148에서 128,898로 82.2%라는 현저한 감소가 일어남을 보였다.

5. 결론

본 논문에서는 삼각망을 이용하여 지형 단순화

Table 1. Experimental results.

정점 추출 개수	평균 오차율(E)		지역 에러 계산 회수	
	실험 1	실험 2	실험 1	실험 2
10	11.20%	11.20%	170,562	30,616
20	7.57%	7.35%	246,699	42,578
30	6.48%	6.51%	281,672	49,674
40	5.97%	6.43%	316,631	55,866
50	6.72%	5.70%	371,430	64,739
60	5.93%	5.15%	388,293	68,450
70	5.51%	5.66%	412,376	72,013
80	4.82%	4.93%	425,435	76,585
90	4.59%	4.81%	437,467	77,902
100	4.44%	4.68%	448,768	79,417
200	3.08%	3.20%	539,719	94,164
300	2.71%	2.47%	591,776	102,716
400	1.84%	1.93%	619,271	109,239
500	1.70%	1.70%	642,522	114,006
600	1.46%	1.52%	660,904	117,746
700	1.40%	1.39%	677,818	121,064
800	1.31%	1.33%	694,966	124,037
900	1.20%	1.23%	706,596	126,114
1000	1.08%	1.09%	718,148	128,898

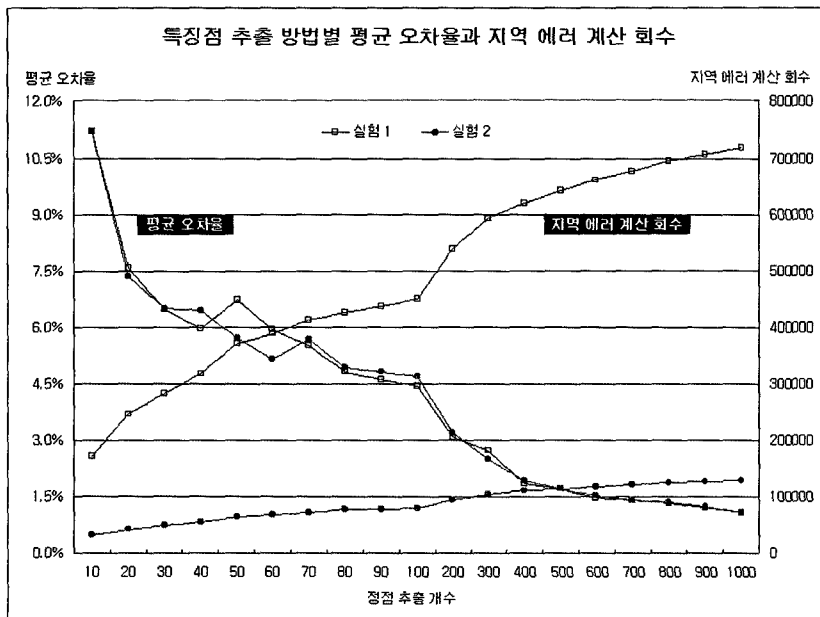


Fig. 7. Error ratio and number of local error computation in terms of the number of vertices.

를 함에 있어서, 정밀성을 손상하지 않으면서 빠른 삼각망 구성을 가능하게 하는 방법을 제안하고 있다. 양질의 삼각망을 구성하기 위해서는, 지형 정보로부터 그 지형을 가장 잘 표현할 수 있는 특징점을 찾아내는 것이 필요하다. 그리디 삽입 알고리즘은 이와 같은 목적을 달성하기 위한 지형 단순화 알고리즘 중의 하나이다.

그러나, 기존의 삽입 알고리즘들의 경우 가장 중요한 특징점을 찾기 위하여 하나의 정점을 추가할 때마다 정해진 탐색공간 내의 모든 정점에 대하여 비교 연산을 수행함으로써 수행 속도에 문제점을 갖고 있다.

본 논문은 이러한 문제점을 해결하기 위하여, 입력된 지형 정보로부터, 지형의 특징을 갖고 있는 능선과 계곡으로 이루어진 특이점들의 집합을 추출하고 이를 상세화의 대상으로 사용함으로써, 전체적인 수행 속도 향상을 얻게 되었다. 지형데이터에 대해서 실험을 실시한 결과 미미한 오차의 증가만으로 현저한 속도의 개선을 얻을 수 있었다.

앞으로 찾아진 능선과 계곡 점들 중에서도 지형을 더욱 잘 반영하는 점들을 선택함으로써 삼각망의 구성을 더욱 가속화하는 방법의 연구가 필요할 것이다.

사 사

이 연구는 2002학년도 단국대학교 대학연구비의 지원으로 연구되었음.

참고문헌

- 강윤식, 박우찬, 양성봉, 2000. 구역화를 이용한 디지털 격자 지형 데이터의 단순화 알고리즘, 한국 정보처리학회 논문지, 7(3): 935-942.
- Floriani, L., B. Falcidieno, and C. Pienovi, 1983. A Delaunay-based method for surface approximation, *Eurographics'83*, pp. 33-35.
- Guibas, L. and J. Stolfi, 1985. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics*, 4(2): 75-123.
- Garland, M. and P. Heckbert, 1995. Fast polygonal approximation of terrains and height fields. Technical report, CS Dept., Carnegie Mellon U., CMU-CS-95-181.
- Guo, D. and D. Richardson, 1998. Automatic vessel extraction from angiogram images, *IEEE Computers in Cardiology*, 25 : 441-444.
- Lawson, C., 1977. *Software for C1 surface interpolation*. In Rice, J., editor, *Mathematical SoftwareIII*, Academic Press.