

가우시안 곡률과 요면을 이용한 메쉬 분할

한태원*, 김선정*

요약

현재 방대한 양의 3차원 물체들이 데이터베이스에 존재하지만 특징별로 검색이나 관리하는 것은 쉽지 않다. 만약 3차원 물체의 모양이 단순하다면 쉽게 특징을 구분 지을 수 있지만 복잡하다면 매우 어려울 것이다. 본 논문은 가우시안 곡률 분석과 오목 정도를 측정하여 3차원 물체의 특징을 분석하고, 특징적인 영역에 대해 한 겹의 경계선을 생성 후 작은 영역들을 융합시키는 메쉬 분할 방법을 제안하고 있다. 이 경계선은 물체의 정확한 특징을 구분하는데 사용되며 구분된 영역들을 고유 특성이 유지되는 방향으로 융합되어 결과물을 산출해 낸다.

Mesh Segmentation Using Gaussian Curvature and Concaveness

Tea-Won Han and Sun-Jeong Kim*

ABSTRACT

There are so many 3D objects in a database but it is not easy to search an object based on its feature or manage it. If the shape of an object is simple, it is not problem to detect features because it is easy to decide standard for mesh segmentation. However, if the shape is complex, it is a big problem. This paper proposes the efficient algorithm for mesh segmentation of 3D objects. We make one-thickness boundary lines based on Gaussian curvature and concaveness, and then segment a mesh using merging small regions to extract exact features. One-thickness boundary lines help us find exact division lines to detect precise features. By applying our algorithm, we could success to segment regions in various models which have a lot of curved faces.

Key Words : Mesh segmentation, Gaussian curvature, Concaveness, Watershed scheme, Region merging

* 한림대학교 컴퓨터공학과

· 제1저자(First Author) : 한태원 · 교신저자(Correspondent Author) : 김선정

· 접수일(2010년 1월 8일), 수정일(1차 : 2010년 1월 29일), 게재확정일(2010년 2월 3일)

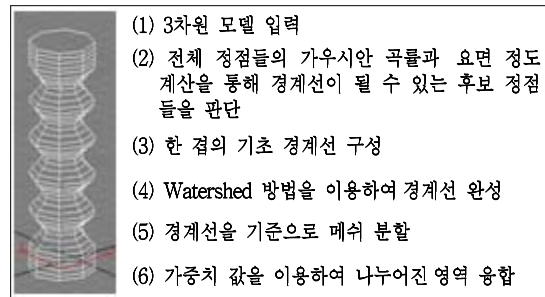
1. 서론

다양한 분야에서 그래픽의 중요성은 좀 더 현실적이고 정확한 분석과 표현을 위해 더욱 강조되어 가고 있다. 실제로 3차원 형태의 객체를 다루는 어플리케이션들이 많이 존재하고 있지만 방대한 양의 데이터베이스에서 필요한 객체를 찾아 활용하기는 쉽지 않다. 그 이유는 사람이 모양의 특징을 인지하는 것과 달리 컴퓨터는 3차원 객체를 하나의 데이터로만 저장하기 때문이다. 하지만 만약 컴퓨터가 사람처럼 사물을 인지할 수 있다면, 인지 기준 또는 특징에 따라 객체들의 유사도를 비교하고 판단할 수 있을 것이다. 그러므로 우리는 많은 3차원 객체 데이터베이스를 유용하게 관리하고 활용할 수 있게 될 것이다. 현재 많은 공업적 모델들은 메쉬 분할을 이용하여 자동으로 그 특징들이 추출되고 있다[1].

본 논문에서 객체의 특징을 찾아 메쉬를 분할하기 위한 과정은 다음과 같다. 첫째, 우리는 각 정점(vertex)의 고유 특징을 구분하기 위해서 가우시안 곡률(Gaussian curvature)과 요면(凹面, concaveness) 정도를 계산해야 한다. 둘째, 메쉬를 분할하기 위해 정확한 경계선을 찾아야 한다. 일반적으로 Watershed 이론은 메쉬 분할의 경계선을 찾는 데 사용되어져왔지만, 이 방법으로 구성된 경계선을 기준으로 메쉬를 나누다 보면 너무 많은 수의 영역들이 생겨난다. 게다가, 전체 정점들의 가우시안 곡률과 요면 정도 값을 고려하지 않기 때문에 정확한 특징을 찾아내기 어렵다. 반면에 우리의 알고리즘은 좀 더 객체의 특징에 적합한 경계선을 찾도록 도와준다. 셋째, 객체의 전체적인 특징을 더 잘 드러내도록 분류된 작은 영역들을 융합(merging)하는 과정이 필요하다. 비록 이전 단계에서 찾아낸 정확한 경계선을 기반으로 메쉬를 나누었지만, 만약 융합하는 과정에서 특징들을 유지시키지 못한다면 올바른 결과를 얻어낼 수 없다. 기존 방법[2]에서는 융합 과정을 수행하기 위해 각각의 영역들이 서로 공

유하고 있는 경계선의 길이를 기준으로 선택하였다. 하지만 이러한 방법은 인접한 영역과의 경계선 길이만을 고려하여 융합되어질 영역을 선택하기 때문에, 융합될 두 영역의 유사정도를 알 수 없으므로 가장 효과적인 융합 방법이라고 할 수 없다. 따라서 본 논문에서는 정점들의 가우시안 곡률과 요면 정도 값을 이용한 융합 과정을 수행함으로써 유사한 영역 간의 결합을 통해 정확한 특징을 찾아낸다. 다음(표 1)은 전체적인 과정을 도표로 표현한 것이다.

표 1. 가우시안 곡률과 요면을 이용한 메쉬 분할 과정
Table 1. Pipeline of mesh segmentation using Gaussian curvature and concaveness



II. 관련 연구

다음은 관련된 가우시안 곡률과 요면 측정, 이론에 대해 간단히 설명한다.

2.1 가우시안 곡률

메쉬를 나누는데 기준이 될 경계선을 구성하기 위해 정점의 곡률을 이용한다. 많은 연구에서 정점의 곡률을 다양한 방법으로 구하여 이용해왔다. 그 중 가우시안 곡률은 3차원 정점의 곡률을 구하는데 매우 효과적인 방법으로, 기하학에서 유래되어 3차원 폴리곤 모델에 적용되어 사용되어왔다[3][4]. Peng et al.[3]과 Gray[4]는 다각형의 3차원 메쉬에 가우시안 곡률을 알

고리즘을 적용시켜 각 정점의 곡률계산에 대하여 다루고 있다. (그림 1)은 3차원 객체의 정점들에서 가우시안 곡률을 계산하기 위한 용어 정의이다.

$$K(v) = \frac{3 \left(2\pi - \sum_{i=0}^{n-1} \alpha_i \right)}{\sum_{i=0}^{n-1} A_i} \quad (1)$$

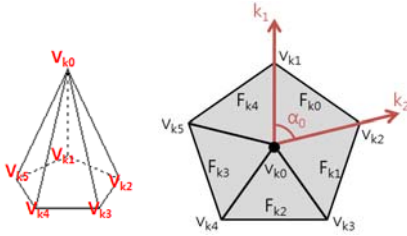


그림 1. 정점 V_{k0} 의 가우시안 곡률 계산을 위한 용어 정의
Fig. 1. Terminology for computing Gaussian curvature of a vertex V_{k0}

V 는 메쉬의 정점들의 집합을 의미하며 각각의 정점 $v_i \in V$ 이다. F 는 메쉬의 면들의 집합을 의미하며 $F_k = \Delta(v_{k0}, v_{k1}, v_{k2}) \in F$ 이다. 한 정점 v_{k0} 을 공유하는 이웃면들 $\{F_{k0}, \dots, F_{kn-1}\} \in F$ 을 구성하는 정점들을 모아 이웃하는 정점들 $\{v_{k1}, \dots, v_{kn}\} \in V$ 을 구할 수 있고, 이로부터 내각을 $\alpha_{k0}, \dots, \alpha_{kn-1}$ 을 계산할 수 있다. 내각 α_{ki-1} 는 모서리 벡터 $k_i = v_{ki} - v_{k0}$ 와 모서리 벡터 $k_{i+1} = v_{ki+1} - v_{k0}$ 의 내적을 이용하여 계산된다.

기하학적인 정의로서, 가우시안 곡률은 표면 S 위에 있는 점 p 를 지나는 무수히 많은 곡선 중에 가장 큰 곡률과 작은 곡률을 갖는 주요 곡률(principal curvature)인 k_1, k_2 의 곱으로, 가우시안 곡률 $K = k_1 \cdot k_2$ 로 표현된다. 그러나 Zang et al.[5]은 폴리곤 메쉬 위에 존재하는 한 정점 v 의 속성으로 사용하기 위해 가우시안 곡률 $K(v)$ 를 다음 공식으로 계산한다.

- 경우 1: $K(v) > 0$ 일 때 정점 v 는 Elliptic
- 경우 2: $K(v) = 0$ 일 때 정점 v 는 Parabolic
- 경우 3: $K(v) < 0$ 일 때 정점 v 는 Hyperbolic

여기서 α_i 는 삼각형 F_i 의 세 내각 중 정점 v 에 대한 내각을 뜻하고, A_i 는 삼각형 F_i 의 면적을 뜻한다. 정확한 가우시안 곡률 값보다는 경우 1~3으로서 형상을 알 수 있기 때문에, $2\pi - \sum_{i=1}^n \alpha_i$ 의 부호 계산만으로 수행 속도를 향상시킬 수 있다. 본 논문에서는 식 (1)을 이용하여 정점 v 의 속성을 구한 후 hyperbolic의 속성을 갖는 정점 v 은 경계선을 구성할 후보 정점이 된다.

2.2 정점의 요면 측정

가우시안 곡률 측정만으로 정확한 경계선을 구성하기 위한 후보 정점을 모두 찾을 수 없다. 그러므로 Schroeder et al.[6]은 정점 v 의 요면 정도를 확인하기 위해 일련의 작업을 수행한다. 정점 v 에 인접하고 있는 이웃면들의 법선 벡터에 대한 가중치 평균인 벡터 n_v 와 이웃하는 정점들 위치에 대한 가중치 평균 v_c 를 이용하여 요면 정도 d 값을 계산할 수 있다. 다음은 d 값을 측정하기 위한 공식을 나타낸다.

$$n_v = \frac{\sum_{i=0}^{n-1} n_i A_i}{\sum_{i=0}^{n-1} A_i} \quad (2)$$

$$v_c = \frac{\sum_{i=0}^{n-1} v_{i+1} A_i}{\sum_{i=0}^{n-1} A_i} \quad (3)$$

$$d = \overrightarrow{v_c v} \cdot n_c \quad (4)$$

다음 단계로는 정확한 3차원 객체의 경계선을 찾기 위해, 기존 방법[2]에서와 같이 가우시안 곡률과 요면 값을 분석한다. 경계선을 구성하는 후보 정점이 되기 위해서는 $K(v) < 0$ 또는 $d < 0$ 을 만족해야 한다. (그림 2)는 요면 측정을 위한 구성요소를 그림으로 표현한 것이다.

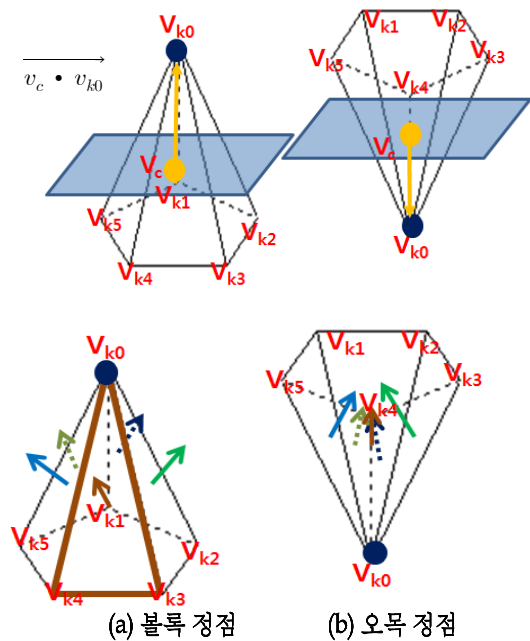


그림 2. 볼록과 오목 정점의 판별을 위한 구성요소
Fig. 2. Components for estimating a convex or concave vertex: (a) convex vertex, (b) concave vertex

III. 1차 경계선의 구성

정점의 특징을 고려하지 않고 단지 Watershed 방법 [7]만을 이용하여 분할된 메쉬 영역의 경계선은 심미적으로 아름답게는 생성되나, 물체 특징에 적합한 경계를 찾아낼 수 없다. 그러므로 본 논문에서는 가우시안 곡률 $K(v)$ 과 요면 정도 d 를 함께 고려하여 물체 특징을 유지할 수 있는 경계선의 후보 정점들을 찾아내고, 한 점으로 구성된 정확한 경계선을 구성하기 위해 일련의 작업을 수행한다.

3.1 외곽선 추출

경계선의 후보 정점들 중에서 우리는 가장 외곽에 위치하고 있는 정점을 찾아 큐(queue)에 저장한다. (그림 3)은 Hose 모델에서 외곽선을 찾는 과정을 보여주고 있다.

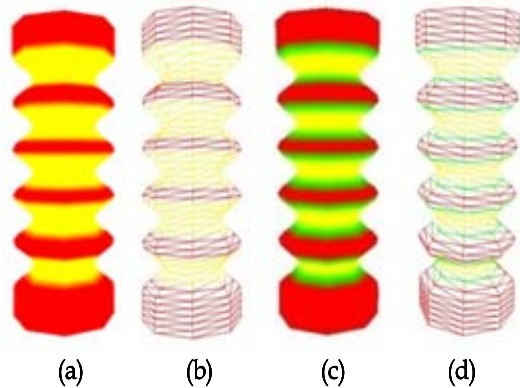


그림 3. Hose 모델의 외곽선 검색과정
Fig. 3. Process of searching the outline of a Hose model

그림 3. (a)와 (b)의 노란색의 정점들은 $K(v)$ 와 d 값을 이용하여 찾아낸 경계선 후보 정점들을 의미한다. (c)와 (d)는 후보 정점들 중 외곽선을 추출하여 녹색으로 표시하였다. 적색은 parabolic 또는 elliptic 속성을 갖는 정점들을 의미한다.

알고리즘 1. 경계선 후보 정점들 중 외곽선 검색

```

for all vertices of boundary candidate
  for visit all neighborhood vertices of
    boundary candidate
      if neighborhood vertex is elliptical or
        parabolic vertex
          push vertex into queue space
        end if
      end for
    end for
  end for

```

3.2 한 겹의 경계선 구성

서로 비슷한 특징을 갖는 정점들을 각각의 영역들로 만들어 정확한 물체의 특징을 추출하기 위해 한 겹의 경계선이 반드시 필요하다. 게다가 만약 경계선의 두께가 한 겹이 아니라면 영역 융합 단계에서 필요한 경계선(서로 다른 영역을 구분하는)을 판단하는데 어려움이 발생할 수 있다. 다음 (알고리즘 2)은 본 논문에서 사용된 한 겹의 경계선 구성 방법을 설명한다.

알고리즘 2. 한 겹의 경계선 구성

```

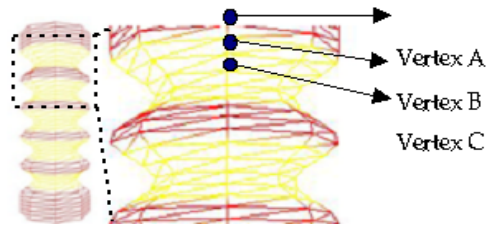
while queue is not empty
  count ← 0 // value for counting
  pop data from queue
  for visit all neighborhood vertices of
    popped vertex
      if attribution of neighborhood vertex is
        boundary candidate AND
        ( value  $d$  of popped vertex  $\geq$ 
          value  $d$  of neighborhood vertex OR
           $K(v)$  of popped vertex  $\geq$ 
           $K(v)$  of neighborhood vertex )
          Increase count value
          push vertex into queue
        end if
      end for
    if count < 2
      attribution of popped vertex ← one-
        thickness boundary line attribution
    else
      attribution of popped vertex ← elliptic
    end if
  end while

```

최종 결정된 한 겹의 경계선을 구성하는 정점들은 이웃하는 인접 정점들 보다 가장 작은 $K(v)$ 또는 d 값을 갖게 된다. 첫째로 외곽선 속성을 갖는 정점들이 저장되어 있는 큐에서 차례로 팝(pop) 되면서 인접 정점들의 $K(v)$ 와 d 값을 비교하는 작업을 수행한다. (표 2)는 Hose 모델의 정점 속성을 보여주며 (그림 4)는 한 겹의 경계선을 찾아가는 과정을 나타낸다. 황색은 경계선 후보 정점들을 나타내며 녹색은 한 겹의 경계선을 나타낸다. 또한 적색은 parabolic 또는 elliptic 속성을 갖는 정점을 뜻한다.

표 2. Hose 모델의 세 정점의 속성 수치
Table 2. Concaveness and Gaussian curvatures for three vertices of a Hose model

Vertex	d	$K(v)$
A	21.900692	-10.391327
B	-9.215116	-16.244385
C	-11.450971	1.067780



IV. 개선된 Watershed 방법에 의한 경계선 완성

본 논문에서는 기존의 Watershed 방법[7]에 한 겹의 경계선 구성을 통한 특징 추출 방법을 추가하여 개선된 Watershed 방법을 제안한다. Watershed 방법은 두 개의 연속된 정점들 간의 정보만을 비교하여 경계선을 찾는 방법으로 bottom-up immersion 접근법과 top-down flooding 접근법 두 종류가 있다. 이 중 bottom-up immersion 접근법은 만약 비교할 정점의 d 값 또는 $K(v)$ 이 인접 정점의 그 값보다 더 크면, 작

은 값을 갖는 인접 정점이 경계선의 속성을 갖게 되는 방식이다. 하지만 본 논문의 제안 방법인 한 겹의 경계선을 구성하는 방법이 적용된다면, 후보 정점들의 속성을 고려하여 경계선을 결정하게 된다. (그림 4)의 1단계~5단계까지의 과정은 1차 경계선 구성과정을 보여주며 6단계에서는 Watershed 방법을 이용하여 경계선을 완성시킨 결과를 나타낸다.

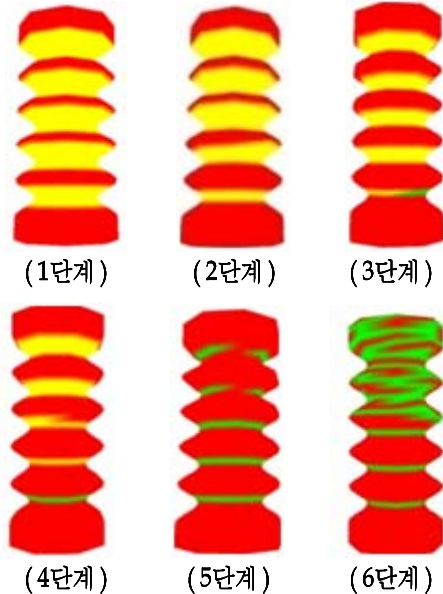


그림 4. Hose 모델의 한 겹의 경계선 구성 과정
Fig. 4. Filtering candidate vertices and finding one-thickness boundary lines

(그림 5)는 기존의 Watershed 방법과 본 논문의 제안하고 있는 개선된 Watershed 방법을 이용하여 구성된 경계선의 결과를 그래프로 도식하여 설명한 것이다.

그림 5. (a)와 (b)는 Watershed 방법의 각각 top-down flooding 접근법과 bottom-up immersion 접근법의 결과이고, (c)는 본 논문이 제안하고 있는 개선된 Watershed 방법의 결과이다. 기존의 Watershed 방법은 메쉬의 특징을 고려하지 않은 경계선을 구성함으로써 구분되어지는 영역의 개수를 불필요하게 증가시키는 경우가 발생한다.

(그림 5)의 경우, (a)는 4개 영역, (b)는 5개 영역으로 나뉘었지만, (c)는 2개 영역으로 나뉘었다.

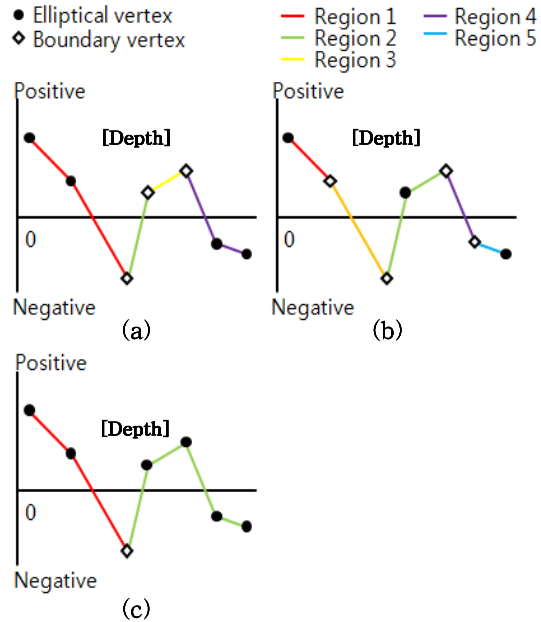


그림 5. 기존의 Watershed 방식(a), (b)와 본 논문의 제안한 개선된 Watershed 방법을 이용한 경계선 결과(c)
Fig. 5. Comparison with the graph results (a) and (b) of Watershed method and our graph result (c) of improved Watershed method

(그림 4)의 5단계에서 볼 수 있듯이 Watershed 방법에서는 시작 정점이 잘못 선택될 경우, 한 겹의 완벽한 경계선을 구성하지 못한다. 개선된 Watershed 방법에서 시작 정점으로 한 겹의 경계선 속성의 이웃 정점을 한 개만 갖는 정점을 선택한다. 이와 같이 선택된 시작 정점이 푸시(push)된 큐를 이용하여 bottom-up immersion 접근법에 기반을 두고 개선된 Watershed 방법은 수행된다.

(그림 6)은 다양한 모델에서 기존의 Watershed 방법을 사용한 결과와 본 논문에서 제안하는 개선된 Watershed 방법을 사용한 결과를 보여준다. (그림 6)에서 검은색은 경계선을 나타내며 각각의 색은 같은

특성을 갖는 정점들의 영역을 나타낸다. (a), (c)는 기존의 Watershed 방법을 사용한 결과이며 (b), (d)는 한 겹의 경계선 방법과 함께 사용되어 개선된 Watershed 방법의 결과를 보여준다.

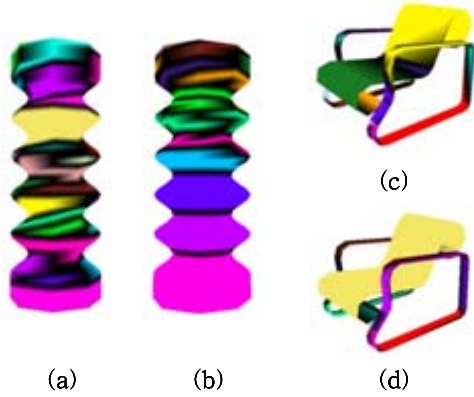


그림 6. 기존 Watershed 방법의 결과 (a), (c)와 개선된 Watershed 방법의 결과 (b), (d)의 비교
Fig. 6. Comparison with the results (a) and (c) of Watershed method and our results (b) and (d) of improved Watershed method

V. 가중치 값을 이용한 영역 융합

나누어진 영역들을 융합시키는 과정은 매우 중요하다. 만약 적합한 영역들끼리 융합되지 않고 무작위로 된다면 고유 특징을 잃기 때문이다. Chen과 Georganas[2]는 융합 알고리즘으로 다음과 같은 과정을 제안하였다. 첫째, 가장 짧은 경계선을 갖고 있는 영역을 찾는다. 가장 작은 경계선을 갖는 영역은 가장 작은 면적을 갖는다고 생각하기 때문이다. 둘째, 그들은 가장 작은 영역을 인접한 영역들 중 하나와 융합한다. 이때 가장 작은 영역을 영역 a, 인접 영역을 영역 b, c, d 라고 할 때 b, c, d 와 a를 구분하는 각각의 경계선이 존재한다. 각각의 경계선 중 가장 긴 경계선을 공유하는 영역과 융합을 하도록 한다. 하지만 비록 경계선의 길이가 길더라도 각 영역의 특징이 비슷하지 않

는 경우가 발생할 수 있다.

따라서 단지 경계선의 길이만을 융합기준으로 수행한다면 전체 객체의 특성을 찾아내지 못할 수 있다. 따라서 본 논문은 두 가지의 융합 기준을 세우고 수행한다. 첫째, 전체 면적의 10% 이하의 영역을 융합이 될 영역들로 구분한다. 이 방법으로 객체에 상관없이 사용자가 분류된 면적의 개수를 각 객체마다 따로 지정할 필요가 없다. 둘째, 영역 간 융합을 수행할 때 경계선의 길이를 사용하지 않고 다음 공식을 이용한다.

$$F = \frac{\sum_{i=1}^n d_i A_i}{\sum_{i=1}^n A_i} \tag{5}$$

A_i 는 영역에 포함된 삼각형의 넓이를 의미하며 d_i 는 영역에 포함된 정점들의 요면 정도를 뜻한다. 결과적으로 식 (5)를 이용하여 가중치 값을 계산하고 이에 기반을 두어 두 영역 간 유사도를 측정한다. (그림 7)은 식 (5)를 이용한 융합 원리를 설명한다. 만약 $F(X_1)$ 가 가장 작은 면적을 갖는 영역이며, $F(X_2) - F(X_1) < F(X_3) - F(X_1)$ 이라면 X_1 은 X_2 에 융합될 것이다.

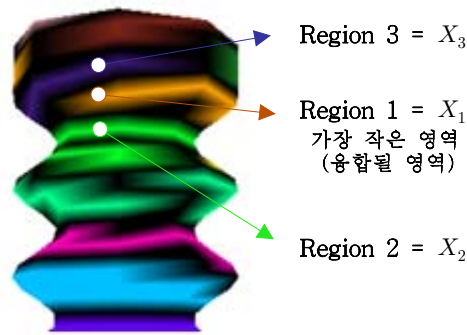


그림 7. 가중치 값을 이용한 영역 융합 원리
Fig. 7. Principal of merging regions based on weights

(그림 8)은 본 논문이 제안하고 있는 가중치 값을 이용한 융합 방법과 기존의 방법인 경계선의 길이를 이용한 융합 방법[2]을 그래프로 도식하여 비교한 것이다. (a)는 샘플 메쉬, C는 가장 작은 영역으로 융합될 영역을 의미한다. (b)는 샘플 메쉬 (a)의 깊이 값을 그래프로 표현한 것이다. (c)는 기존 방법인 경계선의 길이를 이용한 융합 방법의 결과를 나타내며 (d)는 본 논문에서 제안하고 있는 가중치 값을 이용한 융합 방법의 결과를 보여준다. 영역 C는 깊이 값으로 볼 때 영역 A와 유사하므로 두 영역을 융합해야 전체적인 객체의 특징이 계속 유지 될 수 있다. 따라서 본 논문에서 제안하고 있는 융합 작업인 가중치를 사용한 방법이 더욱 효과적 결과를 나타낸다.

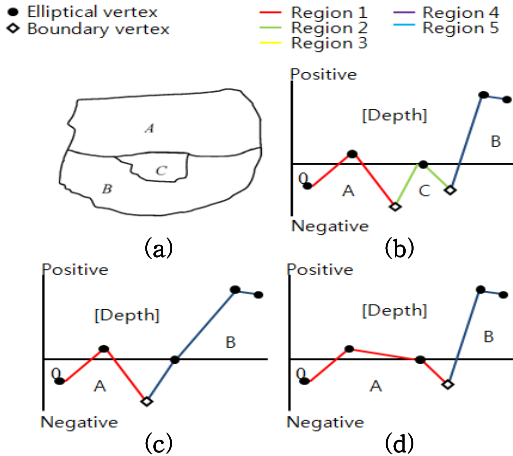


그림 8. 기존 방법인 경계선 길이를 이용한 융합 방법 (c)과 본 논문에서 제안하는 가중치 값을 이용한 융합 방법 (d)의 결과 비교

Fig. 8. Comparison with the graph results (a) and (b) of merging method based on lengths and our graph result (c) of merging method based on weights

V. 결론 및 향후 연구

이 논문은 가우시안 곡률과 요면 정도를 계산하여 3

차원 객체의 특징을 찾은 다음 메쉬 분할의 기준으로 이용하는 방법을 설명하고 있다. 비록 많은 메쉬 분할 방법이 존재하지만 기하학적인 좌표의 특징 분석을 이용하여 다양한 모양을 이루고 있는 객체의 특징을 효과적으로 찾아내기 위해 노력하였다.

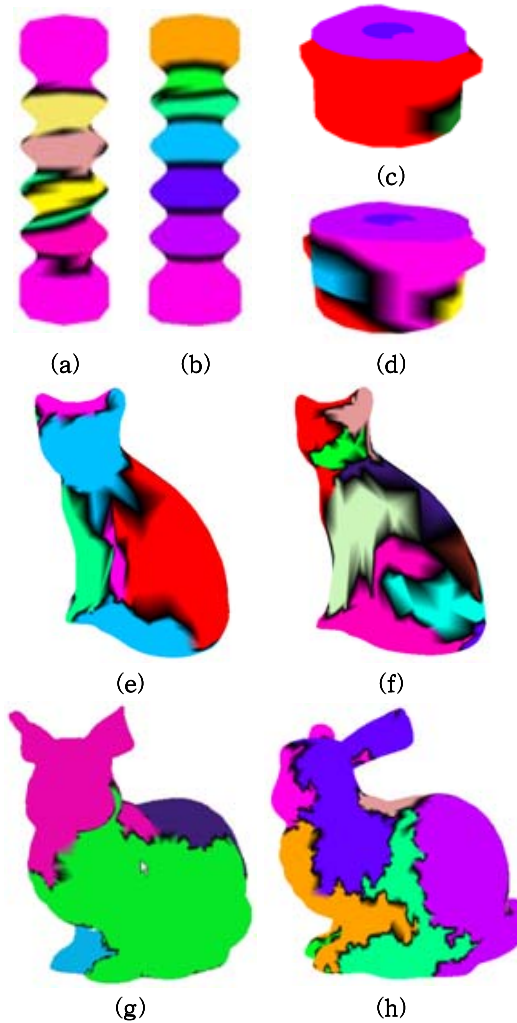


그림 9 기존 방법[2]과 제안 방법의 메쉬 분할 비교:

(a), (c), (e), (g) 기존방법, (b), (d), (f), (h) 제안방법
Fig. 9. Comparison results with a previous method[2] and our proposed method: (a), (c), (e), and (g) are the results of a previous method and (b), (d), (f), and (h) are our results

(그림 9)와 (표 3)은 기존 방법[2]과 본 논문의 제안 방법을 이용한 결과를 비교로, (그림 9)의 (a), (c), (e), (g)는 기존의 방법, (b), (d), (f), (h)는 본 논문에서 제안한 방법의 결과를 보여준다.

표 3. 기존 방법[2]과 제안 방법의 분할 영역 개수 비교
Table 3. Comparison the number of segments with previous method[2] and our proposed method

모델 이름	정점 개수	분할된 영역의 개수	
		제안 방법	기존 방법[2]
Hose	386	9 / 9	
Chair	308	5 / 5	
Car Engine	324	8 / 5	
Cat	1508	11 / 8	
Stanford Bunny	13252	10 / 7	

10% 이하의 면적을 갖는 영역을 융합시킴으로써 모든 모델에서 사용자의 별도의 입력 없이 모든 모델에서 최적화 된 개수의 분할된 영역을 추출하고자 했으나 모든 모델에 동일하게 적용되지는 않았다. 그러므로 본 논문에서 더 나아가 모델에 상관없이 최적화된 특징의 개수를 정할 수 있는 기준을 세운다면 객체에 특징에 맞는 완전 자동의 메쉬 분할이 가능할 것이다.

향후 연구로 본 논문의 3차원 메쉬 분할 알고리즘을 기계 학습에 활용할 수 있을 것이라 예상된다.

Acknowledgement

본 연구는 2009년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구입니다. (2009-00765445)

참고 문헌

[1] Sunil V. B. and Pande S. S, "Automatic recognition of features from freeform surface CAD models", Computer-Aided Design, Vol. 40, No. 4, pp.502-517, 2008.
[2] Chen L and Georganas N. D, "An efficient and robust algorithm for 3D mesh segmentation", Multimedia Tools

Appl., Vol. 29, No. 2, pp. 109-125, 2006.
[3] Peng J, Li Q, Kuo C-C. J, and Zhou M, "Estimating Gaussian Curvatures from 3D Meshes", Human Vision and Electronic Imaging VIII. Edited by Rogowitz, Bernice E.; Pappas, Thrasyvoulos N. Proceedings of the SPIE, Volume 5007, pp. 270-280, 2003.
[4] Gray A, "The Gaussian and mean curvatures `14.5 in modern differential geometry of curves and surfaces", CRC, Boca Raton, Florida, pp. 279-285, 1993.
[5] Zhang Y, Paik J. K, Koschan A, Abidi M. A, and Gorsich D, "A simple and efficient algorithm for part decomposition of 3D triangulated models based on curvature analysis", Proc. Int. Conf. Image Processing, Rochester, New York, Vol. III, pp. 273-276, 2002.
[6] Schroeder W, Zarge J, and Lorensen W, "Decimation of triangle meshes", Computer Graphics 25(3):65-70 (Proc. SIGGRAPH '92), 1992.
[7] Pulla S, Razdan A, and Farinz G, "Improved Curvature Estimation for Watershed Segmentation of 3-Dimensional Meshes", IEEE Trans Vis Comput Graph, 2001.
[8] 안성욱 외, "DirecX와 인공지능을 효율적으로 사용한 3D 바둑게임", 한국지식정보기술학회 논문지, 제4권 제2호, pp.45-52, 2009.



한태원(Tea-Won Han)

2006년 한림대학교 컴퓨터공학 학사

2009년~현재 한림대학교 산학협력단 인턴연구원
※ 관심분야: 영상인식, 3차원 메쉬 처리, 인공지능



김선정(Sun-Jeong Kim)

1996년 고려대학교 컴퓨터학과 학사
1998년 고려대학교 컴퓨터학과 석사
2003년 고려대학교 컴퓨터학과 박사

2005년~현재: 한림대학교 컴퓨터공학과 부교수
※ 관심분야: 컴퓨터 그래픽스, 3차원 게임, 가상현실