

논문 2005-42SP-6-16

모델 단순화를 위한 오차 측정 방법

(Error measure for vertex simplification)

김 윤 정*, 이 병 옥*

(Yoonjung Kim and Byung-Uk Lee)

요 약

이동 통신이나 인터넷 채널과 같이 전송용량이 제한적인 환경에서 3차원 물체에 그래픽의 사용이 빈번해 지면서 3차원 모델을 단순화시켜야 할 필요성이 증대되고 있다. 삼각형으로 표현된 3차원 물체의 표면을 단순화시키는 경우, 3차원 공간에서의 꼭지점들 중에서 제일 가까운 점 두 개를 선택하여 반복적으로 데이터를 줄여가는 vertex contraction 방법이 많이 사용되고 있다. 이때 단순화 된 점의 위치를 결정하기 위하여 단순화시 발생하는 오차를 계산하여야 한다. 본 논문에서는 새로운 오차 계산법을 제시하여 원 모델과의 오차가 작게 단순화하는 방법을 제안하고, 이를 3차원 모델 데이터에 적용하여 실제 개선결과를 확인한다.

Abstract

Simplification of 3D models is becoming necessary with popularity of 3D graphics over mobile or the internet channels with limited channel capacity. Surfaces of a 3D model are usually approximated by a series of triangular meshes, and vertex contraction method is employed widely to minimize the deviation from the original model. Determination of the best position after contraction depends on the calculation of simplification error. We propose a new measure for computing the error so that the simplified model represents the original faithfully. We demonstrate the improved results with real 3D models.

Keywords : simplification, error measurement

I. 서 론

컴퓨터 그래픽을 사용하는 게임, 애니메이션 등이 널리 상용화 되어감에 따라서 3차원 모델의 사실에 대한 기대가 커지고 있다. 사용자들은 영상의 사실성 (reality) 을 위하여 모델이 자세히 묘사되고, 표면의 특성, 즉 색상과 질감 같은 성질도 정확하게 표현되기를 바란다. 그러나 사실성이 증가할 수록, 모델에 대한 데이터 량이 증가하고 이에 따라 이동통신 환경에서의 사용에 제한이 생기고, 실시간 실행이 어려워진다. 이 문제를 해결하기 위하여 물체를 구성하고 있는 3D 꼭지점 정보를 압축

하여 전송과 표현에 사용하기 위한 많은 연구가 이루어지고 있다. 최근의 3D 모델의 단순화는 삼차원 모델을 표현하고 있는 표면을 단순화하는 방법 (surface simplification) 에 대한 연구가 주류를 이루고 있다. 지금까지 발표된 표면 단순화에 대한 여러 가지의 알고리즘들은 다음과 같은 3개의 범주로 분류할 수 있다. 꼭지점 제거 (vertex decimation) 방법^[1]은 국소 형태 방법 (local shape heuristics) 에 따라 가장 중요도가 낮은 꼭지점을 선정하여 그 점을 포함하는 모든 면을 제거한 후에 재삼각형화 하는 과정을 반복함으로써 꼭지점의 개수를 줄여가는 방법이다. 이 방법으로 좋은 결과를 얻을 수 있지만 매니폴드 (manifold) 표면에만 제한 될 수밖에 없는 꼭지점 분류 (classification) 와 재삼각형과의 구성을 사용하므로 단순화 과정에서 물체의 위상 기하 (topology) 를 단순화시킬 수 없는 단점이 있다. 또한 이 방법은 다해상도

* 정회원, 이화여자대학교 정보통신학과
(Ewha Womans University Information Electronics)
* 본 연구는 대학 IT연구센터 육성지원사업의 연구결과로 수행되었음.
접수일자: 2005년4월7일, 수정완료일: 2005년9월22일

렌더링 시스템 (multiresolution rendering system)에 제한적인 요소가 되기 때문에 우리의 실험에 적합하지 않다. 꼭지점 합체 (vertex clustering) 방법^[2]은 원 모델의 가장자리 (boundary)를 구한 후에 그 안을 격자로 나누어서 각각의 칸 안에 위치하고 있는 모든 점들을 단 하나의 꼭지점으로 합친 후에 그에 따라 물체의 표면을 새로 구성하는 방법으로, 매우 빠르게 수행되지만 격자의 칸의 크기가 위치적인 에러 경계 (geometric error bound)의 원인이 되기 때문에 단순화된 결과의 화질이 낮은 경우가 종종 발생한다. 게다가 한 격자의 칸의 크기와 위치 변경에 대해 무척 민감하기 때문에 실험에 대한 제어가 용이하지 않다. 반복적 선분 압축 (Iterative edge contraction)^[3,4,5,6]은 두 점을 이은 각각의 선분을 조건에 따른 비용에 의해 등급을 매겨서 등급이 가장 낮은 선분에 대해 압축 (contraction)을 실행하고 그 선분에 연결되어 있는 면의 정보를 수정하는 방법이다(그림 1). 이 방법은 조건에 따라서 단순화 과정에서 구멍을 매우거나 연결되어 있지 않은 점 두개를 합침으로써 모델의 위상 기하학을 단순화시키기도 한다. 이 방식을 기본으로 하는 여러 알고리즘들의 가장 큰 차이점은 압축을 적용시킬 선분을 정하는 방법으로, 가장 대표적인 예는 Hoppe^[7], Ronfard and Rossignac^[8], Guziec^[9]의 알고리즘이 있다. 이 방법은 대체로 좋은 결과를 제공하고, 실험에 필요한 제어가 가능하기 때문에 논문에서는 이 방법을 사용하여 단순화를 실행하였다. 이 방법의 대부분의 알고리즘^[5,6]에서는 프로세스 실행 과정 중에 계산해야 할 단순화 에러에 대하여 이차 오류 함수 (quadric error metric)를 정의하여 사용하였다. 급격한 곡선이나 얇은 부분을 가지고 있는 모델에 대해서 이산 미분 함수 (discrete differential metrics)를 사용하는 알고리즘^[3]과 모델에 대한 전체적인 구조적 시점에서의 중요성을 결정하고

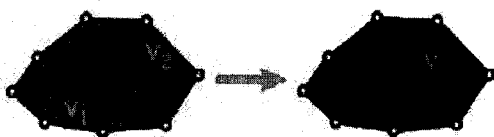


그림 1. 선분으로 연결된 두 점을 한점으로 줄이는 선분 압축

Fig. 1. Edge contraction method.

그에 따라 단순화를 수행하는 알고리즘^[4] 등의 여러 가지 방식의 오차 함수 정의가 연구되고 있다.

이 논문에서는 반복적 선분 압축을 사용하여 단순화를 수행 하였을 때, 목표로 정해진 두 점의 압축 과정에서 중요하게 사용되는 단순화 오류를 이전의 알고리즘보다 더 정확하게 계산하여 더 좋은 단순화 결과를 만들 수 있는, 3차원 모델에 대한 거리계산법에 새롭게 적용되는 오류 계산법을 제안한다. 또한 원 모델과 단순화를 행한 모델간의 차이 오류를 계산하는 과정에서 한 방향으로 비교하는 것이 아니라 두 모델 간의 상호비교를 통하여 두 모델 간의 유사성을 더 보존할 수 있도록 개선된 알고리즘을 제안한다.

II. 본 론

1. 기존의 단순화 알고리즘

지금까지 발표된 논문 중에서 가장 널리 쓰이고 있는 반복적 선분 압축을 토대로 한 꼭지점 제거 알고리즘은 이차 오류 함수를 사용하여 두 목표점들을 대체하는 새로운 꼭지점의 위치를 결정하는 방식이다^[5,6]. 이차 오류 함수는 각 꼭지점에 대하여 그 점을 포함하고 있는 면들의 법선 벡터를 이용하여 각 면들의 기초 오류 이차 행렬 (fundamental error quadric matrix)을 구한 후, 이들의 더하는 것으로 정의하였다. 이 행렬은 한 점과 그 점과 관련된 면들 간의 최단 거리의 제곱 합을 계산한다. 단순화 과정에서 두 점이 합쳐져서 새로 만들어지는 점의 위치를 정할 때 거리 제곱의 합이 가장 작은 값을 가지는 위치의 점을 결정하기 때문에 단순화 후의 최적의 위치를 정하는 척도로 사용한다. 알고리즘의 기본적인 구조에 대해 간단히 설명하면 다음과 같다. 우선 3차원 모델을 표현하고 있는 선분 중에서 실험에 제일 적합한 조건을 가지고 있는 한 쌍의 점 (vertex pair)을 결정해야 한다. 이 알고리즘에서는 원 모델과 가장 유사한 단순화 결과를 얻는 것이 목적이므로 가장 짧은 거리에 있으며 선분으로 연결되어 있는 두 점을 단순화 대상으로 결정하고 이 한 쌍의 점을 valid pair라고 정의하였다. 이 쌍은 압축이 실행된 후에는 \bar{v} 의 새로운 위치를 가지게 되므로 압축 시에는 v_1 뿐만 아니라 v_2 에 연결되어 있는 선분에 대한 모든 정

보가 필요하다. 이 때, \mathbf{v}_1 과 \mathbf{v}_2 의 위치를 $\bar{\mathbf{v}}$ 로 수정한다. $(\mathbf{v}_1, \mathbf{v}_2)$ 의 새로운 위치인 $\bar{\mathbf{v}}$ 는 비용 값이 가장 적은 위치로 결정한다. $\bar{\mathbf{v}}$ 은 \mathbf{v}_1 의 위치와 \mathbf{v}_2 의 위치, 그리고 $(\mathbf{v}_1 + \mathbf{v}_2)/2$ 의 위치에서 각각 비용을 계산한 후에 그 중에서 가장 적은 비용을 가지는 위치로 정해진다. 비용은 모델을 이루고 있는 각각의 점에서 정해지는 값으로 압축 후에 원 모델에서 얼마나 바뀌는 가를 나타내는 오류 값으로 정의한다. 이 오류 값은 \mathbf{v}_1 과 \mathbf{v}_2 가 세 후보 위치중 하나의 위치 $\bar{\mathbf{v}}$ 로 통합되었을 때 생기게 되는 오류, 즉 그 점들을 포함하였던 모든 면들과 $\bar{\mathbf{v}}$ 와의 거리제곱의 합이다. 이는 \mathbf{v}_1 과 \mathbf{v}_2 를 포함하는 면들에서 계산한 기초 이차 행렬(fundamental quadric matrix) 들을 모두 합한 이차 행렬 \mathbf{Q} 를 사용하여 구한다. 각 후보 위치 \mathbf{v} 의 비용 값은

$$\Delta(\mathbf{v}) = \mathbf{v}^T \mathbf{Q} \mathbf{v} \tag{1}$$

의 식을 사용하며, 압축 후의 새로운 위치 $\bar{\mathbf{v}}$ 의 이차 행렬은

$$\mathbf{Q}(\bar{\mathbf{v}}) \cong \mathbf{Q}(\mathbf{v}_1) + \mathbf{Q}(\mathbf{v}_2) \tag{2}$$

의 단순한 덧셈으로 값으로 근사된다.

2. 개선된 단순화 알고리즘

앞 절에서 설명하고 있듯이, 하나의 점이 감소되어 지는 한 번의 단순화의 대상 점 두 점의 새로운 위치를 결정하기 위하여 미리 정하여 놓은 방식으로 각 점에 대한 비용을 계산하게 된다. 이 때, 한 점의 비용은 그 점과 연관되어 있는 면들과의 거리를 모두 합한 값으로, 그 점이 제거되었을 때 발생할 것이라고 예상되는 오류 값으로 사용된다. 따라서 이 점과 면들 간의 거리의 합의 정확성은 단순화 과정에 매우 결정적인 역할을 한다. 제안한 알고리즘은 거리계산과정의 정확성을 높이기 위한 새로운 계산 방법으로, 점과 면의 위치를 조사하여 이에 따라서 거리 측정을 달리하는 방식으로 개선되었다. 또한 두 점 중에서 한 점이 제거되었다고 가정하였을 때의 오류를 계산하는 과정에서, 기준을 단순화 실행 전의 모델로만 잡아서 실행 후의

모델과 비교하여 오류를 결정하는 것뿐만 아니라, 실행 후의 모델을 기준으로 실행 전의 모델의 오류를 계산하여 함께 사용함으로써 오류의 정확성을 높일 수 있는 단계를 삽입하였다.

가. 개선된 오류 측정 방법

이전의 알고리즘에서는 앞에서 설명한 바와 같이 한 쌍의 점을 단순화 하는 과정에서 오류 수치를 결정하기 위한 측정방법으로 이차 오류 함수를 사용하였다. 그러나 지금까지 사용해 온 이차 오류 함수는 한 점과 면간의 거리를 계산할 때, 그 점에서 면을 포함하는 평면에 내린 수선의 발이 항상 내부에 있다는 가정을 하고 계산하기 때문에 수선의 발이 외부에 존재할 때의 경우를 생각하지 않는다. 이에 따라서 이를 고려하기 위하여 오류를 계산하는 방법을 수정하였다. 즉, 더 정확한 오류 값을 계산하기 위하여 이차 오류 행렬 알고리즘을 사용하지 않고 다음과 같은 방법으로 대체하였다. 한 점과 평면 간의 최단거리를 구하기 위해서는 점에서 평면으로 내린 수선의 발과 점과의 거리를 측정하면 된다. 그러나 점의 수선의 발이 항상 평면의 내부에 존재하지 않기 때문에 이때에는 평면과 가장 가까운 위치에 있는 선분이나 점을 찾아서 이 거리를 측정해야 한다. 다시 말하여 기존의 방법에서는 한 점에서 삼각형 평면에 내린 수선의 길이를 계산하기 때문에 그 수선의 발이 삼각형 외부에 있는 경우 실제 보다 오차가 작은 것으로 나타나서 잘못된 결과를 만들어 낸다. 제안된 방법에서는 수선의 발이 삼각형 외부에 있는 경우도 정확하고 빠

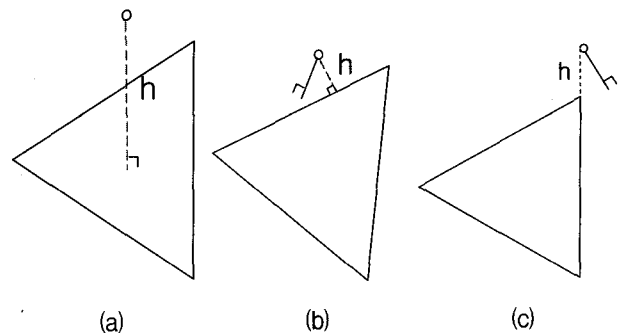


그림 2. (a) 수선의 발이 삼각형 내부에 있는 경우, (b) 최단 거리 점이 삼각형 선분에 있는 경우, (c) 최단 거리 점이 삼각형 꼭지점인 경우

Fig. 2. (a) The nearest point is inside the triangular patch, (b) on the edge of the triangle, (c) on the vertex.

르게 계산할 수 있어서 모델을 단순화시킨 후에 원 모델과의 차이를 줄일 수 있다.

대상 점에서 면을 포함하는 평면에 수선의 발을 내렸을 때, 1) 수선의 발이 면을 이루는 삼각형의 내부에 있는 경우, 그 유클리드 거리 (euclidean distance)를 구할 수 있다. (그림 2 (a)) 수선의 발이 삼각형의 외부에 있는 경우에는 다음의 두 가지 경우로 분류할 수 있다. 2) 점이 삼각형의 어느 한 변에 가까이 있다면 삼각형과 두 점 간의 최단 거리는 그 변에 대해 내린 수선의 발까지의 거리이다. (그림 2 (b)) 마지막 경우로 3) 점에서부터 선분까지 수선의 발을 내릴 수 없다면 삼각형의 세 점 중 한 점과의 거리를 가장 짧은 거리로 결정하는 것이다. (그림 2 (c)) 이 세 가지 경우를 고려하여 단순화를 하면 더 정확한 오류 값을 가지고 점을 조정할 수 있다.

경우 1) P_1, P_2, P_3 의 세 점이 이루는 평면 위의 점 P 는 계수 α, β, γ 을 사용하여 다음과 같이 나타낼 수 있다.

$$P = \alpha P_1 + \beta P_2 + \gamma P_3 \tag{3}$$

이 때, $\alpha, \beta, \gamma = 1$ 이고, α, β, γ 의 각각의 값에 의하여 점 P 가 P_1, P_2, P_3 의 삼각형 내부에 있는지를 판단할 수 있다. 즉, α, β, γ 가 $0 < \alpha < 1, 0 < \beta < 1, 0 < \gamma < 1$ 의 범위에 있으면 점 P 는 삼각형의 내부에 존재한다.

따라서 대상 꼭지점인 점 X 에서 삼각형을 포함하는 평면에 대하여 수선을 내렸을 때, 그 수선의 발을 점 P 라 하고, 위의 식 (3)의 형태로 점 P 를 나타내어 점 P 가 삼각형의 내부에 존재하는지를 판단한다. 이 때, 점 P 가 삼각형의 내부에 있을 조건을 만족한다면 라그랑지 곱수 (Lagrange multiplier)를 사용하여, 점 X 와 점 P 간의 거리 $\|X - P\|$ 가 최소가 되는 점 P 의 좌표, 즉, α, β, γ 의 값을 계산한다.

$$\begin{aligned} f(\alpha, \beta, \gamma) &= (X - P)^t (X - P) + \lambda(\alpha + \beta + \gamma - 1) \\ &= \alpha^2 P_1^2 + \beta^2 P_2^2 + \gamma^2 P_3^2 + 2\alpha\beta P_1 P_2 + 2\beta\gamma P_2 P_3 + 2\gamma\alpha P_3 P_1 \\ &\quad - 2X(\alpha P_1 + \beta P_2 + \gamma P_3) + X^2 + \lambda(\alpha + \beta + \gamma - 1) \end{aligned} \tag{4}$$

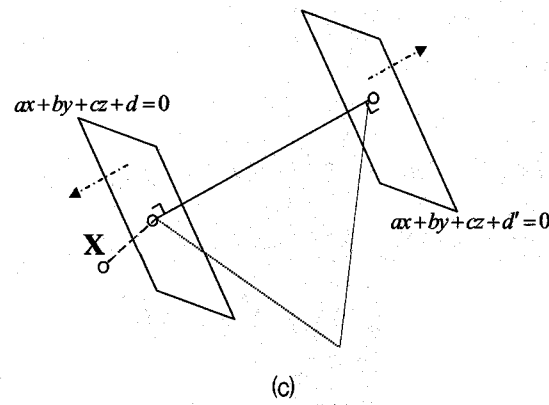
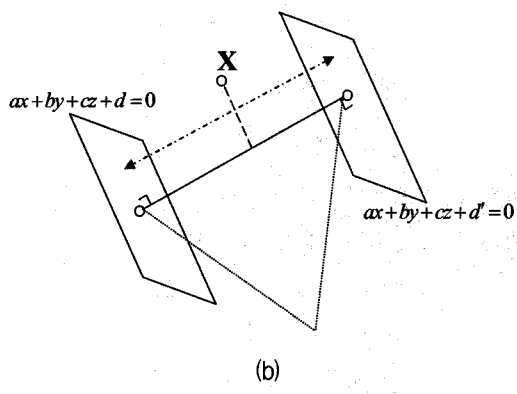
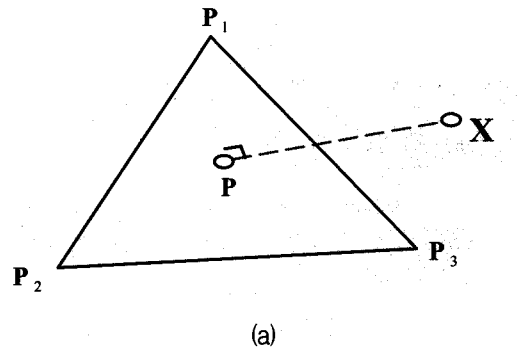


그림 3. 대상 점과 삼각형 사이의 거리 계산
(a) 최단 거리 점이 삼각형 내부에 있는 경우
(b) 최단 거리 점이 삼각형 선분인 경우 (c) 최단 거리점이 삼각형 꼭지점인 경우

Fig. 3. The distance between a point and the triangle
(a) The nearest point is inside the triangle. (b) The nearest point is on the edge of the triangle. (c) The nearest point is a vertex of the triangle.

$$\begin{aligned} \frac{\partial f}{\partial \alpha} &= 2\alpha P_1^2 + 2\beta P_1 P_2 + 2\gamma P_3 P_1 - 2P_1 X + \lambda = 0 \\ \frac{\partial f}{\partial \beta} &= 2\beta P_2^2 + 2\alpha P_1 P_2 + 2\gamma P_2 P_3 - 2P_2 X + \lambda = 0 \\ \frac{\partial f}{\partial \gamma} &= 2\gamma P_3^2 + 2\alpha P_3 P_1 + 2\beta P_2 P_3 - 2P_3 X + \lambda = 0 \end{aligned} \tag{5}$$

$$\begin{pmatrix} 2P_1^2 & 2P_1P_2 & 2P_3P_1 \\ 2P_1P_2 & 2P_2^2 & 2P_2P_3 \\ 2P_3P_1 & 2P_2P_3 & 2P_3^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 2P_1X - \lambda \\ 2P_2X - \lambda \\ 2P_3X - \lambda \end{pmatrix}$$

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 2P_1^2 & 2P_1P_2 & 2P_3P_1 \\ 2P_1P_2 & 2P_2^2 & 2P_2P_3 \\ 2P_3P_1 & 2P_2P_3 & 2P_3^2 \end{pmatrix}^{-1} \begin{pmatrix} 2P_1X - \lambda \\ 2P_2X - \lambda \\ 2P_3X - \lambda \end{pmatrix} \quad (6)$$

위의 식에서 $\alpha, \beta, \gamma = 1$ 의 제한조건이 있기 때문에 미지수 λ 를 구할 수 있고, λ 는 α, β, γ 의 값을 결정하므로 이로써 점 P 의 위치를 얻을 수 있고, 이에 따라 점 X 와 삼각형의 최단거리를 계산할 수 있다.

경우 2) 점 X 의 수선의 발인 점 P 가 삼각형의 외부에 있다고 판단되면 삼각형을 이루는 세 선분과 점 X 의 관계를 결정한다. 각 선분에 대하여 그 선분을 법선 벡터로 가지는, 선분의 끝 점을 지나 는 두 개의 평면의 방정식을 찾아, 이 두 평면의 사이에 점 X 가 존재한다면 우리가 원하는 점 X 와 면간의 최단거리는 점 X 에서 그 선분에 대하여 수선의 발을 내렸을 때의 거리가 되는 것이다.

경우 3) 두 평면의 외부에 점 X 가 존재한다면, 삼각형의 세 점과의 거리 중 가장 짧은 거리를 계산한다.

나. 결정기준 오류의 비교대상

앞에서 설명한 바와 같이 단순화 수행 하는 과정은 최단 선분을 찾아서 이 선분의 두 점을 한 점으로 줄이는 것이다. 기존의 알고리즘에서, 새로운 점의 위치를 결정하는 데에 기준이 되는 각 점에서의 예상 발생 오류는 점이 줄어든 상황을 가정하여 이 모델과 그 단순화 단계를 거치기 전의 모델의 차이점을 계산하였다. i 번의 단순화 과정을 거친 후의 모델을 M_i 라고 하자. M_i 에서는 원 모델에서 i 개의 꼭지점이 줄어든 것이다. M_i 에서 한 쌍의 점에 대한 새로운 점 \bar{v} 의 후보 위치는 기존의 두 점의 위치와 두 점의 중앙 위치가 된다. 지금까지의 알고리즘은 $(i-1)$ 번의 단순화 후의 모델 M_{i-1} 이 세 후보 위치에 대해서 각각 그 위치로 단순화 후의 모델이 정해지면 발생하게 되는 M_i 과의 차이점을 그 위치 점에 대한 오류 값이라 하고 이 값이

가장 작은 위치로 \bar{v} 의 위치를 결정하는 방법을 사용하였다. 즉, i 번 째 단순화 과정을 실행하기 위해서는 $(i-1)$ 번째 단순화 결과 모델을 기준으로 하고, 새로운 점 \bar{v} 의 변화에 따른 오류를 사용한다. 따라서 이 오류 값들은 $(i-1)$ 번째 한 단계 앞의 단순화된 모델에 대하여 얼마나 변하는 것인지만 고려하게 된다.

이는 i 번째 단순화의 결과인 M_i 이 한 번의 단순화과정 전의 모델인 M_{i-1} 에 대해서만 얼마나 유사성을 가지는지를 고려하여 \bar{v} 의 위치를 결정하는 것으로, M_{i-1} 을 기준으로 하여 M_i 에 대하여 가지게 되는 오류 값은 고려하지 않은 결과를 가지고 단순화를 수행함을 나타낸다.

이 문제를 보완하기 위하여 M_{i-1} 에서 \bar{v} 이 세 위치 중에 어느 한 위치로 결정 되었다고 가정하였을 때의 M_i 에 대한 오류를 계산하여 같이 사용하였다. 그림 4에서 보는 것과 같이, M_{i-1} 의 꼭지점에서 M_i 까지의 최단 거리를 구하여 계산한 오류(점선)와 M_i 의 단순화된 꼭지점에서 M_{i-1} 까지의 최단거리를 구한 오류(실선)는 서로 다른 값을 가진다. 따라서 단순화 위치를 선정하는 기준인 오류 값을 계산하는 경우 이 두 오류 값을 고려하여 \bar{v} 의 위치를 선정함으로써 좀 더 전체 오류를 적게 나타낼 수 있는 M_{i-1} 에 비슷한 M_i 을 만들어 낼 수 있다. 실험에서는 두 개의 모델 사이에 계산된 2개의 오류 중에서 큰 오류를 선택하는 것으로 실

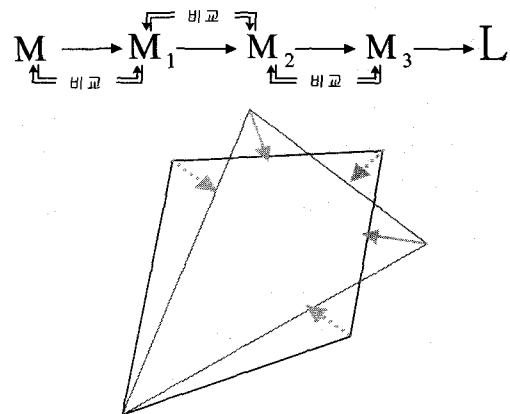


그림 4. 양방향으로 오류를 결정하는 경우
Fig. 4. Error calculation considering both directions.

행하였다.

III. 실험

이 논문에서 사용된 3D 모델의 형식은 VRML V.2.0을 사용하였다. 실험 영상은 36866개의 꼭지점, 73728개의 면을 가진 컬러 지구 영상과 35947개의 점, 69451개의 면을 가진 단색 Stanford bunny영상을 사용하였다.

가. 결과 영상

단색 bunny 모델을 가지고 실험을 하였을 경우, 단방향 오류계산에 의한 결과와 양방향 오류계산에 대한 결과가 동일하였다. (그림 5)

컬러 모델인 지구 영상을 사용하였을 때의 결과는 그림 6에서 보듯이 비슷한 계열의 색상을 가지고 있는 부분의 색의 분포도를 비교하여 보았을 때, 양방향 오류 계산 결과 영상이 보다 더 원영상과 흡사한 것으로 나타났다. 이는 단순화 실행 과

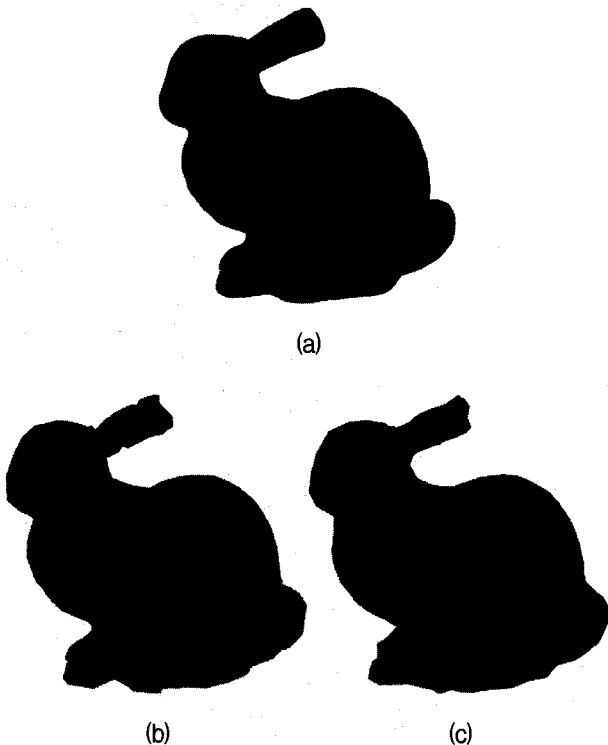


그림 5. 실험에 사용된 bunny 영상
 (a) 원 모델 (35947점, 69451면)
 (b) 기존 알고리즘 적용 (2000점, 1725면)
 (c) 제안된 알고리즘(2000점, 1725면)

Fig. 5. Bunny model.
 (a) Original model (b) Current algorithm
 (c) Proposed algorithm

정에서 위치적 오류보다 색상 오류가 더 민감한 요소라는 것을 확인할 수 있다. 표 1은 본론의 2절 2항에서 제안한 새로운 오류 측정 방법을 적용하였을 때의 개선정도를 나타낸 것이다. 모든 점과 면 사이의 거리에 대하여 무조건 점의 수선의 발이 면의 내부에 있다고 가정하는 기존의 알고리즘 방법 대신에, 새로운 방법으로 오류를 분류하였을 때 청계산횟수의 약 88%의 경우에서 보다 더 정확한 오류 계산결과를 만들어 내었다. 이는 실험 모델에 따른 오차가 존재하겠지만 대부분의 거리계산에서 대상 점과 면 사이의 거리의 최단거리가 수선의 발이 아니기 때문에 이 논문에서 제안한 것과 같은 새로운 알고리즘을 사용하여 정확한 단순화 결과를 얻을 수 있음을 증명한 것이다. 표 2는 각 실험 모델에 대하여 결과영상과 원영상 간의 오류를 계산한 것이다. 이 오류는 앞 장에서 설명한, 두 점을 한 점으로 압축하는 과정에서 생기는 이전 모델과의 차이점을 모두 더한 것이다. 표의 수치로도 확인할 수 있듯이, 오류를 계산하는 과정을 좀더 세분하여 정확성을 올리기 위한 개선된 알고리즘이 시각적인 개선뿐만 아니라 전체 오류 값에서도 더 작은 오류수치를 나타낸다. 단색 bunny 영상에서는 모델의 총 꼭지점 개수가 35947개에서 2000개

표 1. 새로운 알고리즘의 적용정도
 Table 1. The application of the proposed algorithm.

계산 횟수 모델	기존의 알고리즘이 적용되는 계산횟수	새로운 알고리즘이 적용되는 계산횟수
bunny 영상	281616 (10.12%)	2501472 (89.88%)
지구 영상	234588 (8.64%)	2479236 (91.36%)

표 2. 기존의 알고리즘과 개선된 알고리즘의 오류 비교
 Table 2. error comparison between the existing and proposed algorithms.

방법 모델	기존의 알고리즘	단방향	양방향
bunny	4.34×10^{-5}	3.65×10^{-5}	3.65×10^{-5}
지구	2.56×10^{-5}	2.03×10^{-5}	1.98×10^{-5}

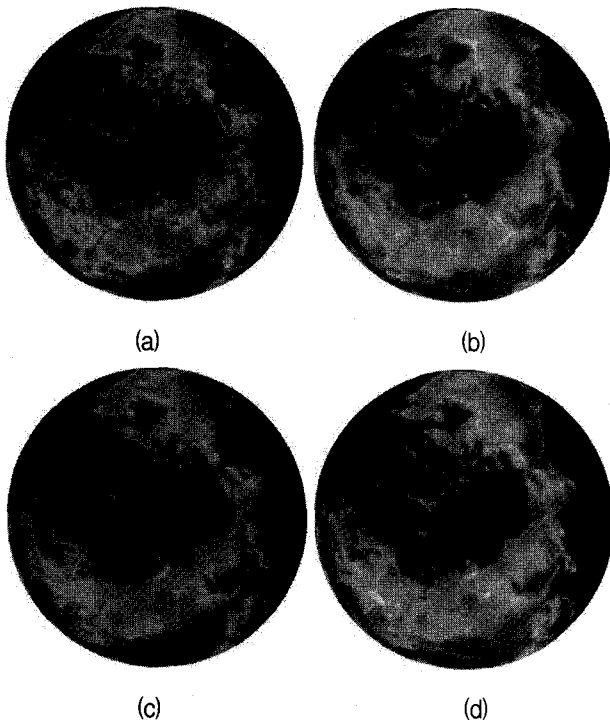


그림 6. 실험에 사용된 지구 영상
 (a) 원영상 (36866점, 73728면)
 (b) 기존의 알고리즘 결과영상 (5000점, 9944면)
 (c) 단방향으로 오류를 계산하여 단순화를 실행한 결과영상 (5000점, 9907면)
 (d) 양방향으로 오류를 계산하여 단순화를 실행한 결과영상 (5000점, 9907면)

Fig. 6. Globe Model.
 (a) Original model (b) Current algorithm
 (c) Proposed monidirection error algorithm
 (d) Proposed bidirection algorithm

로 줄어들었을 때, 개선된 알고리즘으로의 단순화 실행 후의 오류가 기존의 알고리즘으로 단순화를 실행한 후의 오류보다 약 16%가 감소하였고, 지구 영상에 대한 비교실험에서는 36866 개의 꼭지점에서 5000개의 꼭지점으로 단순화 하였을 때에, 오류 수치가 단방향 오류 측정 시에는 약 21%, 양방향 오류 측정 시에는 약 23%의 오류가 감소하는 결과를 얻을 수 있었다.

IV. 결 론

본 논문에서는 3D 물체의 전송이나 처리를 용이하게 하기 위한 과정인 꼭지점 정보의 단순화에 대하여 모델 표면의 단순화 결과가 기존의 알고리즘 결과보다 더 원 모델과 유사하게 단순화를 실행하는 새로운 알고리즘을 제안하였다. 수정된 알고리

즘은 합쳐지는 두 꼭지점의 새 위치를 결정하는 오류 계산 방법을 계산 결과 값이 더 정확한 값을 가지도록 세분화하였으며, 오류를 계산하는 과정에서 오류의 비교대상을 바꾸어 주면서 양방향으로 오류 계산을 하여서 그 중의 최대 오류를 취함으로써 오류의 정확도를 높이도록 하였다. 이 알고리즘과 기존의 알고리즘을 비교해본 결과, 시각적인 차이와 수치적인 해석을 통한 결과 모두에서 수정된 알고리즘이 이전의 알고리즘보다 원래 영상과 단순화된 영상 사이의 오류를 적게 만들어냄을 확인하였다.

참 고 문 헌

- [1] William J. Schroeder, Jonathan A. Zarge, and William E. Lorensen, "Decimation of triangle meshes," *Computer Graphics (SIGGRAPH '92 Proc.)*, Vol. 26, No. 2, pp. 65-70, July 1992.
- [2] Jarek Rossignac and Paul Borrel, "Multi-resolution 3D approximations for rendering complex scenes," *Modeling in Computer Graphics: Methods and Applications*, pp. 455-465, June 28-July 2 1993.
- [3] Sun-Jeong Kim, Soo-Kyun Kim, and Chang-Hun Kim. "Discrete differential error metric for surface simplification," *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings*, pp. 276 - 283, Oct. 2002.
- [4] Jingqi Yan, Pengfei Shi, and David Zhang, "Mesh Simplification with Hierarchical Shape Analysis and Iterative Edge Contraction," *IEEE Trans. on Visualization and Computer Graphics, Vol. 10, No. 2*, pp. 142 - 151, Mar/Apr 2004.
- [5] Michael Garland and Paul S. Heckbert, "Simplifying surface with color and texture using quadric error metrics," *IEEE Visualization 98 Conference Proceedings*, pp. 263-269, October 1998.
- [6] Michael Garland and Paul S. Heckbert, "Surface simplification using quadric error metrics," *SIGGRAPH 97 Proc.*, pp. 209-216, August 1997.
- [7] Hugues Hoppe, "Progressive meshes," *SIGGRAPH 96 Proc.*, pp. 99-108, Aug. 1996.
- [8] Remi Ronfard and Jarek Rossignac, "Full-range approximations of triangulated polyhedra," *Computer Graphics Forum, Eurographics '96, Vol. 15, No. 3, Proc.* Aug. 1996.
- [9] Andre Guziec, "Surface simplification with variable tolerance," *Second Annual Intl. Symp.*

On Medical Robotics and Computer Assisted Surgery (MRCAS '95), pp. 132-139, November 1995.

저 자 소 개



김 윤 정(정회원)
 2002년 이화여자대학교 정보통신
 학과 학사 졸업.
 2004년 이화여자대학교 정보통신
 학과 석사 졸업.
 2004년 이화여자대학교 정보통신
 학과 시간강사.

<주관심분야 : 영상처리, 그래픽>



이 병 옥(정회원)
 1979년 서울대학교
 전자공학과 학사 졸업.
 1981년 KAIST 전기 및
 전자공학과 석사 졸업
 1991년 Stanford University Dept.
 of Electrical Engineering
 박사 졸업.

1981년~1995년 대우전자 영상연구소 수석연수원
 1995년~현재 이화여자대학교 정보통신학과 교수
 <주관심분야 : 영상처리, 영상압축, 영상인식>