

블렌드쉐이프를 위한 다수 삼각 메쉬의 동시 단순화 기법

박정호^{1O} 김종용² 송중훈³ 박상훈⁴ 윤승현^{5*}

^{1,5}동국대학교 멀티미디어공학과 ^{2,3,4}동국대학교 영상대학원 멀티미디어학과

wjdgh283@hanmail.net khj8499@dongguk.edu whdrktsk1218@naver.com mshpark@dongguk.edu shyun@dongguk.edu

Simultaneous Simplification of Multiple Triangle Meshes for Blend Shape

Jung-Ho Park^{1O} Jongyong Kim² Jonghun Song³ Sanghun Park⁴ Seung-Hyun Yoon^{5*}

^{1,5}Dept. of Multimedia Engineering, Dongguk University ^{2,3,4}Dept. of Multimedia, Dongguk University

요약

본 논문에서는 동일한 정점의 수와 연결 정보를 갖는 N 개의 삼각 메쉬의 형상을 동시에 단순화하는 새로운 기법을 제시한다. 기존의 단순화 기법을 N 개의 삼각 메쉬에 각각 적용하면, 정점의 개수는 같지만 삼각형을 구성하는 연결 정보가 서로 다른 단순화 메쉬를 생성한다. 이러한 한계는 고해상도 블렌드쉐이프(blend shape) 모델에서 단순화된 블렌드쉐이프 모델을 구성하는 것을 어렵게 한다. 본 논문에서 제시된 기법은 N 개 메쉬 형상을 동시에 고려하여 최소의 제거 비용을 갖는 에지를 선택하여 단순화를 수행한다. 따라서 단순화 결과로 생성된 N 개의 단순화 메쉬는 동일한 정점의 수와 연결 정보를 유지하게 된다. 다양한 얼굴 표정을 갖는 고해상도 블렌드쉐이프 모델에 동시 단순화 기법을 적용하여 단순화된 블렌드쉐이프 모델을 생성함으로써 제안된 기법의 효율성과 유효성을 입증한다.

Abstract

In this paper we present a new technique for simultaneously simplifying N triangle meshes with the same number of vertices and the same connectivities. Applying the existing simplification technique to each of the N triangle mesh creates a simplified mesh with the same number of vertices but different connectivities. These limits make it difficult to construct a simplified blend-shape model in a high-resolution blend-shape model. The technique presented in this paper takes into account the N meshes simultaneously and performs simplification by selecting an edge with minimal removal cost. Thus, the N simplified meshes generated as a result of the simplification retain the same number of vertices and the same connectivities. The efficiency and effectiveness of the proposed technique is demonstrated by applying simultaneous simplification technique to multiple triangle meshes.

키워드: 메쉬 단순화, 블렌드쉐이프, 얼굴 애니메이션

Keywords: Mesh Simplification, Blend Shape, Facial Animation

1. 서론

컴퓨터그래픽스 기술의 비약적인 발전으로 영화 및 애니메이션 제작에 가상의 3D 캐릭터의 등장은 새로운 일이 아니다. 국내 및 해외의 영화 업계에서 작품을 살펴보면 과거와 비교할 수 없을 정도의 많은 CG장면과 그 속에 등장하는 3D 캐릭터의 숫자는 점점 증가하는 추세이다 [1]. 얼굴 표정은 영화나 애니메이션에서 캐릭터의 특징을 나타내기 위한 중요한 수단이기 때문에 더욱

사실적이고 풍부한 얼굴 애니메이션을 만들기 위한 기술이 실제 산업에서 사용되어 왔다. 블렌드쉐이프(blend shape) 기법은 고품질의 3D 얼굴 애니메이션을 생성하기 위해 가장 널리 사용되는 방법이며, 특히 영화 업계에서 현실적인 휴머노이드 캐릭터를 만들기 위해 주로 사용된다 [2]. 블렌드쉐이프는 무표정의 베이스쉐이프(base shape)의 형상을 변형시켜 다른 표정의 타겟쉐이프(target shape)을 생성하고, 두 형상간의 선형보간을 통해 새로운 얼굴 표정을 합성하는 기술이다. Figure 1처럼 무표정 형상을 가진 베

*corresponding author: Seung-Hyun Yoon/Dongguk University(shyun@dongguk.edu)

스케입이 있다고 하자. 정점의 개수와 연결구조가 동일한 복사본 모델의 입술 양끝 정점을 움직여서 웃는 형상을 가진 타겟스케입을 얻으면 두 개의 모델은 형상은 다르지만 정점의 개수와 연결 정보가 동일하기 때문에 모든 정점에 대해 일대일 대응관계를 생성할 수 있다. 즉 두 모델은 정점의 개수, 각 정점의 인덱스 뿐만 아니라 정점들간의 연결 정보가 서로 동일하다. 따라서 무표정 모델의 정점을 시간에 따라 웃는 얼굴 모델의 정점 위치로 이동시켜 얼굴 애니메이션을 생성할 수 있다.

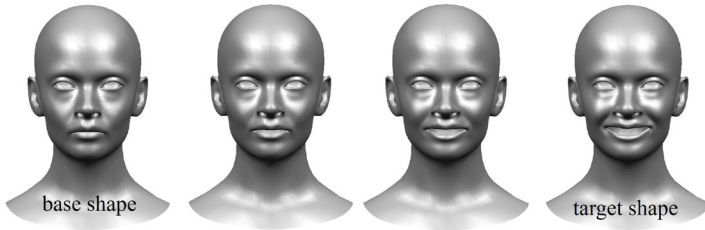


Figure 1: The process of creating a blend shape by linearly interpolating a base shape and a target shape.

얼굴 모델이 수백만개의 폴리곤으로 이루어져 있는 경우 고성능 그래픽스 하드웨어에서조차 이를 처리하기 위한 비용이 크다. 따라서 얼굴 애니메이션을 수행하기 전에 각각의 모델을 단순화할 필요가 있다. 하지만 정점의 개수와 연결 정보가 모두 동일한 얼굴 모델 집합에서 각각의 모델을 단순화했을 때 정점의 연결 정보가 서로 달라지는 문제가 발생한다. 결국 모델 정점간의 일대일 대응관계가 사라지기 때문에 더이상 단순화된 모델간에 블렌드스케입 기법을 적용하기 어려워진다. 본 논문에서는 이러한 문제를 해결하기 위해 다수의 삼각 메시에 대하여 정점의 개수와 연결 정보를 유지하는 동시 단순화 기법을 제안한다.

본 논문의 구성은 다음과 같다. 제 2절에서는 블렌드스케입 및 메시 단순화와 관련된 이전 연구를 소개한다. 제 3절에서는 기존에 널리 사용되는 단일 메시의 단순화 기법을 설명하고, 제 4절에서는 확장된 에지 제거 정보를 이용하여 단일 메시의 단순화 기법을 다수의 메시로 확장한다. 제 5절에서는 다양한 실험을 통해 제안된 기법의 안정성과 유효성을 보이고, 끝으로 제 6절에서는 결론 및 향후 연구방향을 제시한다.

2. 관련 연구

2.1 블렌드스케입을 이용한 얼굴 애니메이션

지금까지 다양한 얼굴 애니메이션 기술이 개발되어 왔다. 그 중 전통적인 방식인 키프레임 사이에 선형 보간법을 이용하여 중간 프레임을 생성하는 방식은 Parke와 Frederick [3]가 처음으로 제안하였다. Lewis 등 [4]은 블렌드스케입 기법을 위상적으로 일치하는 여러 형상의 선형 가중치 합으로 설명하면서 슬라이더 인터페이스를 이용하여 가중치를 조절하는 예를 보여주었다. 타겟스케입의 모델수가 많거나 모델의 삼각형 수가 많은 경우 얼굴 애

니메이션을 가속화하기 위한 몇몇 연구도 진행되었다. Seo 등 [5]은 대용량의 블렌드스케입 모델을 압축하여 하드웨어 가속 애니메이션을 수행하는 방법을 다루었으며, 행렬 블록 재정렬(roworder reordering)과 HSS (hierarchically semi-separable) 표현에 기반한 행렬 압축 기법을 제시하였다. Costigan 등 [6]은 GPU에 블렌드스케입 모델을 저장함으로써 속도가 향상되는 애니메이션 기법을 제시하였다. 아직까지는 얼굴 애니메이션 처리를 가속화하기 위한 타겟스케입을 동시적으로 단순화하는 연구는 진행되지 않은 상황이다.

2.2 메시 단순화

본 논문에서 제시하는 알고리즘의 목표는 다수의 폴리곤모델을 동시에 단순화하며 위상을 같게 유지하는 것이다. 지금까지 많은 종류의 메시 단순화 알고리즘이 제안되었다. Schroeder 등 [7]은 국소적 지오메트리를 검사하면서 정점을 점진적으로 제거하였다. [8]에서는 메시표면 위에 새로운 정점을 생성하고 곡률이 높은 영역에서 많은 정점을 분포시키며 메쉬를 재구성(re-tiling)하는 방법을 제시하였다. 이 방식은 날카로운 모서리를 가진 메시보다는 비교적 매끄러운 표면으로 이루어진 메시에 적합하다. Calvin과 Taylor [9]는 삼각형을 병합하는 방식으로 단순화를 수행했다. 이 방식은 얼굴 모델보다는 날카로운 모서리를 많이 가진 모델에 더 적합하다. Garland와 Heckbert [10]가 처음으로 2차 에러 측정법을 이용하여 폴리곤 모델을 단순화하는 알고리즘을 제시하였다. [11]에서는 메시의 볼륨이 선형적으로 변화하며 단순화가 진행된다. Cignoni 등 [12]은 이전 알고리즘의 성능을 분석 비교한 결과를 제시하였다. 메시 단순화 알고리즘의 정확도는 메시의 기하학적 구조나 위상정보 등에 따라 크게 좌우된다. 이후 Pauly 등 [13]은 곡률이 높은 영역에 더 많은 정점을 유지시키기 위해 국소 변이 추정과 2차 에러 측정법을 사용하였다. 그러나 이 기법은 메시 단순화를 포인트 클라우드 기반에서 수행한다는 점에서 멀티레벨 스무딩, 다중 해상도 모델링과 같은 포인트 기반 프로세싱 어플리케이션에 더 적합하다. Kim 등 [14]은 곡률이 높고 형상이 얇은 영역에서는 2차 에러 측정법이 정확한 기하학적 오차를 측정하는 것이 어렵다는 점을 보완하여 날카로운 형상의 영역을 보존하기 위한 단순화 기법을 제안하였다. Ran 등 [15]이 제시한 방법은 메시의 날카로운 특징을 잘 보존한다는 장점이 있지만 지역적 들로네(delaunay) 조건이 유지되어야 한다는 점에서 제한적이다. 본 논문에서는 많은 문헌에서 소개된 2차 에러 측정법을 이용한 메시 단순화를 간략히 소개하고, 다수의 삼각 메시에 대해 확장하는 기법을 제안한다.

3. 단일 메시의 단순화

본 절에서는 Garland와 Heckbert [10]가 제안한 단일 삼각 메시에 대한 단순화 기법을 요약한다. 제 4절에서는 이를 다수의 삼각 메시에 대한 동시 단순화 기법으로 확장한다.

3.1 정점의 QEM 행렬

3차원 공간의 한 점 $\mathbf{v} = [x \ y \ z]^T$ 에서 임의의 평면 $\pi: ax + by + cz + d = 0$ 까지 거리의 제곱(squared distance)은 다음과 같이 2차식으로 표현될 수 있다:

$$\begin{aligned}\delta(\mathbf{v}) &= (\mathbf{p}^T \mathbf{v})^2 \\ &= \mathbf{v}^T (\mathbf{p} \mathbf{p}^T) \mathbf{v} \\ &= \mathbf{v}^T K \mathbf{v},\end{aligned}$$

여기서 $\mathbf{p} = [a \ b \ c \ d]^T$ 는 단위법선($a^2 + b^2 + c^2 = 1$)을 갖는 평면 π 의 동차좌표(homogeneous coordinates) 표현이고, K 는 다음과 같이 계산되는 대칭행렬이다:

$$K = \mathbf{p} \mathbf{p}^T = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} \quad (1)$$

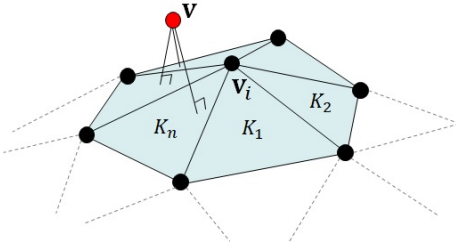


Figure 2: Sum of squared distances between a vertex and planes.

Figure 2에서 삼각 메쉬 위의 한 정점 \mathbf{v}_i 를 공유한 n 개의 삼각형을 가정하자. 메쉬 외부의 한 점 \mathbf{v} 에서 삼각형을 포함한 각 평면까지의 거리 제곱의 합은 다음과 같이 계산될 수 있다:

$$\Delta_i(\mathbf{v}) = \mathbf{v}^T K_1 \mathbf{v} + \mathbf{v}^T K_2 \mathbf{v} + \dots + \mathbf{v}^T K_n \mathbf{v} = \mathbf{v}^T Q_i \mathbf{v},$$

여기서 K_j 는 각 삼각형을 포함한 평면으로부터 식 (1)을 통해 계산되는 행렬이며, $Q_i = \sum_{j=1}^n K_j$ 는 정점 \mathbf{v}_i 에서 정의되는 QEM (quadratic error metric) 행렬이라고 한다. 메쉬 단순화 기법에서 행렬 Q_i 는 한 점 \mathbf{v} 에서 정점 \mathbf{v}_i 주변의 메쉬까지의 형상의 차이(에러)를 측정하기 위해서 사용된다.

3.2 메쉬 단순화 알고리즘

삼각 메쉬에서 반복적으로 에지를 제거(collapse)함으로써 단순화된 메쉬를 얻을 수 있다. Figure 3은 \mathbf{v}_1 과 \mathbf{v}_2 를 양 끝점으로 하는 에지 $e = (\mathbf{v}_1, \mathbf{v}_2)$ 를 제거하는 과정을 나타낸다. 에지 제거 과정은 (1) \mathbf{v}_1 과 \mathbf{v}_2 를 \mathbf{v}' 으로 병합하고, (2) \mathbf{v}_2 와 연결된 에지를 \mathbf{v}_1 에 연결하고, (3) 사용되지 않는 삼각형을 제거하는 단계로 구성된다.

에지 제거로 단순화된 메쉬(Figure 3(b))는 제거 전 메쉬(Figure 3(a))와 형상의 차이를 갖는다. 이러한 형상의 차이는 에지 제거에 따른 비용으로 정의될 수 있으며, 이는 \mathbf{v}' 에서 제거 전 메쉬의 삼각형을 포함한 평면들까지의 거리 제곱의 합으로 다음과 같이 계산될 수 있다:

$$C(\mathbf{v}') = \mathbf{v}'^T Q_1 \mathbf{v}' + \mathbf{v}'^T Q_2 \mathbf{v}' = \mathbf{v}'^T (Q_1 + Q_2) \mathbf{v}',$$

여기서 Q_1 과 Q_2 는 각각 정점 \mathbf{v}_1 과 \mathbf{v}_2 에서 계산된 QEM 행렬이다. 에지 제거 비용은 정점 \mathbf{v}' 의 위치에 따라 달라지므로 최소의 제거 비용을 갖는 \mathbf{v}' 의 위치는 $\frac{\partial C}{\partial x} = \frac{\partial C}{\partial y} = \frac{\partial C}{\partial z} = 0$ 를 만족하는 해 $(\hat{x}, \hat{y}, \hat{z})$ 를 구하여 찾을 수 있다.

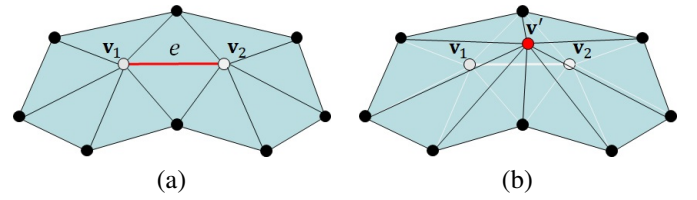


Figure 3: Edge collapse operation.

메쉬를 구성하는 각각의 에지 e_i 에 대하여 제거 위치와 비용으로 구성된 에지 제거 정보 $(e_i, \mathbf{v}'_i, C_i)$ 를 구하여 최소힙(min heap)에 저장한다. 최소힙은 에지 제거 비용이 가장 적은 에지를 루트에 유지하므로, 최소힙에서 순차적으로 에지 제거 정보를 꺼내어 해당 에지를 제거하면 원본 삼각 메쉬와 형상의 차이를 최소화하는 단순화 메쉬를 얻을 수 있게 된다. Algorithm 1은 단일 메쉬에 대한 단순화 알고리즘을 나타내고, Figure 4는 얼굴 메쉬에 단순화 기법을 적용한 결과를 나타낸다.

Algorithm 1: Simplification of single triangle mesh

Input: M : input mesh, $vnum$: target number of vertices
Output: M' : simplified mesh
for each vertex $\mathbf{v}_i \in M$ **do**
 | compute Q_i ;
end
for each edge $e_i \in M$ **do**
 | compute collapse position \mathbf{v}'_i and cost C_i ;
 | push $(e_i, \mathbf{v}'_i, C_i)$ into min-heap;
end
while # of vertices $> vnum$ **do**
 | pop $(e_i, \mathbf{v}'_i, C_i)$ from min-heap;
 | collapse e_i into \mathbf{v}'_i ;
 for each edge e_j incident to \mathbf{v}'_i **do**
 | compute collapse position \mathbf{v}'_j and update cost C_j ;
 | push $(e_j, \mathbf{v}'_j, C_j)$ into min-heap;
 end
end

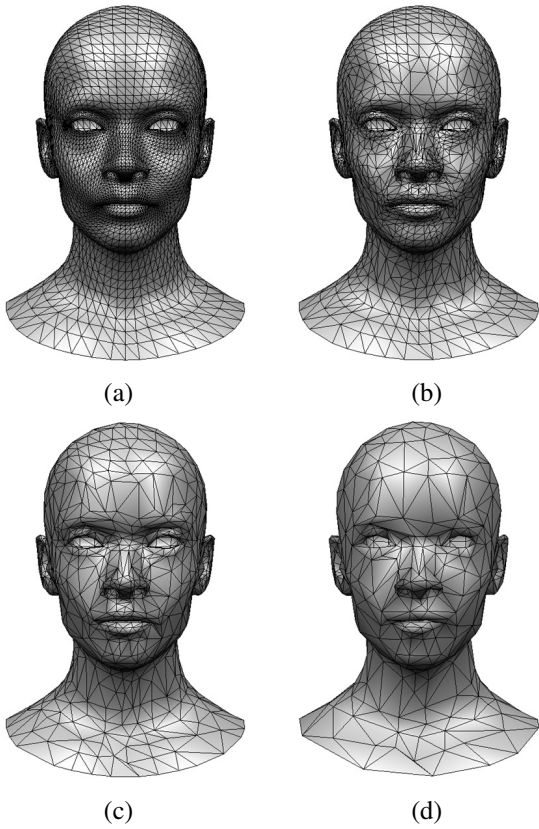


Figure 4: Results of mesh simplification: (a) original mesh, (b) 50% simplification, (c) 25% simplification and (d) 12% simplification.

4. 다수 메쉬의 동시 단순화

본 절에서는 앞에서 설명한 단일 메쉬의 단순화 기법을 확장하여 동일한 연결 정보를 갖는 N 개의 삼각 메쉬에 동시에 적용하는 기법을 제안한다. 제안된 알고리즘은 N 개의 메쉬를 동시에 단순화할 때, 에지 제거 순서를 동일하게 적용하여 최종적으로 얻어진 단순화 메쉬들이 동일한 정점의 수와 연결 정보를 갖도록 한다.

4.1 동시 단순화 가능 메쉬

일반적으로 블렌드셰입 모델은 중립(neutral) 표정을 갖는 메쉬에서 변형된 다양한 표정을 갖는 메쉬들로 구성된다. 블렌드셰입을 구성하는 메쉬들은 하나의 메쉬에서 변형되었기 때문에 동일한 정점의 수와 연결 정보를 공유한다. 다양한 표정들을 갖는 메쉬들을 가중치를 기반으로 선형결합을 하면 새로운 표정을 갖는 메쉬를 생성할 수 있다. 본 논문에서 제안하는 동시 단순화 기법은 이러한 제약 조건 즉, 동일한 정점의 수와 연결 정보를 갖는 다수의 메쉬들을 대상으로 한다. Figure 5은 본 논문에서 사용한 블렌드셰입 모델의 일부 메쉬들을 나타낸다.

4.2 확장된 에지 제거 정보

제 3절에서 정의한 에지 제거 비용은 에지 주변의 메쉬 형상에 따라 달라진다. 따라서 N 개의 메쉬가 동일한 정점의 수와 연결 정보를 갖더라도, 메쉬의 기하학적 형상이 다르면 각 에지의 제거 비용은 메쉬에 따라 서로 다르게 된다. 결과적으로 기존의 메쉬 단순화 기법을 블렌드셰입을 구성하는 각각의 메쉬에 개별적으로 적용할 경우, 제거되는 에지의 순서가 메쉬마다 다르기 때문에 최종적으로 단순화된 메쉬들은 서로 다른 연결 정보를 갖게 된다. 이러한 결과 메쉬들에 대하여 블렌드셰입 기법 즉, 가중치를 이용한 선형 결합을 적용할 경우 원하지 않는 결과를 초래하게 된다. 본 논문에서는 이러한 문제를 해결하기 위해 N 개의 메쉬의 형상을 동시에 고려하여 확장된 에지 제거 정보를 구성하고, 이를 기반으로 모든 메쉬에서 동일한 순서로 에지를 제거하는 기법을 제시한다.

동일한 정점의 수와 연결 정보를 갖는 N 개의 메쉬 집합 $\mathcal{M} = \{M^1, M^2, \dots, M^N\}$ 을 가정하자. 메쉬 M^j 의 에지 e_i 의 제거 비용을 C_i^j 라고 하면, 모든 메쉬를 고려한 에지 e_i 의 제거 비용 C_i 는 다음과 같이 계산될 수 있다:

$$C_i = \sum_{j=1}^N w_j C_i^j,$$

여기서 w_j 는 메쉬 M^j 에 대한 에지 e_i 의 제거 비용에 대한 가중치이다. 모든 메쉬에 대하여 가중치가 $w_j = 1$ 인 경우, 에지 e_i 의 제거 비용은 단순히 모든 메쉬에서 해당 에지의 제거 비용을 합한 결과와 같다. 본 논문에서는 에지 e_i 의 제거 비용의 가중치를 각 메쉬마다 다르게 적용한다. 적절한 가중치 w_j 를 설정하기 위해 각각의 메쉬 M^j 에 대하여 에지 e_i 의 제거 비용 C_i^j 를 계산한다. 만약 제거 비용 C_i^j 가 모든 메쉬 모델에 대하여 동일하거나 거의 유사하다면, 이러한 결과는 에지 e_i 주변에서 각 메쉬의 형상이 크게 변형되지 않기 때문이다. 반면, 에지 e_k 는 메쉬 모델에 따라 제거 비용 C_k^j 의 변화가 크다고 하면, 이는 에지 e_k 주변에서 각 메쉬의 형상의 변화가 크다는 것을 유추할 수 있다. 제거 비용의 변화가 큰 에지 e_k 는 e_i 에 비해 상대적으로 각 메쉬들에서 형상의 중요한 특징을 표현한다고 해석할 수 있다. 따라서 이러한 에지들은 제거 비용을 더 크게하여 단순화 과정에서 제거되지 않도록 한다.

각 메쉬에 대하여 에지 e_i 의 제거 비용의 표준 편차(standard deviation)를 σ_i 라고 하자. σ_i 가 작은 에지 e_i 는 각 메쉬에서 형상의 변화가 적은 부분에 포함되는 에지이므로 낮은 가중치를 할당하여 에지 제거의 순위를 높일 수 있다. 반면, 제거 비용의 표준 편차가 큰 에지는 높은 가중치를 할당하여 해당 에지가 제거되지 않도록 할 수 있다. 이러한 원리를 적용하기 위해 본 논문에서는 다음과 같은 표준 편차에 따른 가중치 함수를 이용하였다:

$$w(\sigma) = \begin{cases} \frac{0.9}{\sigma_{max}} \times \sigma + 0.1 & \text{if } 0 \leq \sigma \leq \sigma_{max}, \\ 1 & \text{otherwise,} \end{cases}$$

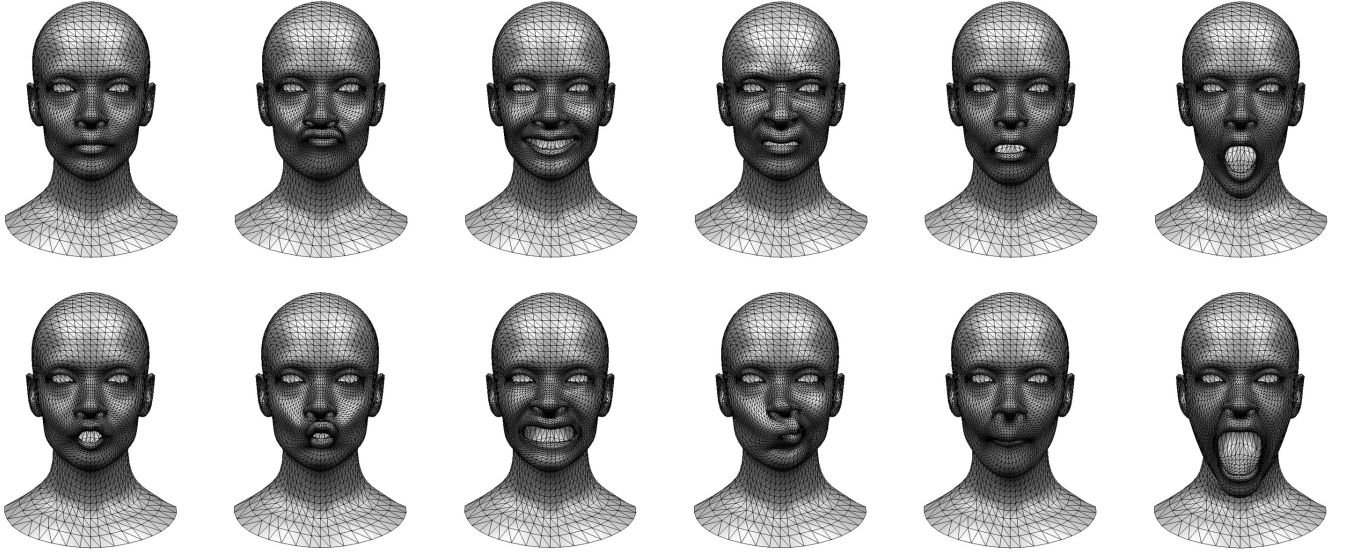


Figure 5: A set of mesh models for blend shape.

여기서 σ_{max} 는 사용자가 정한 에지 제거 비용의 최대 편차를 나타내고, 가중치는 $0.1 \leq w \leq 1$ 의 범위에 포함된다. 확장된 에지 제거 정보는 Figure 6와 같이 각 모델에 대한 에지 제거 비용과 가중치 그리고 모든 메쉬를 고려한 제거 비용으로 구성된다. Algorithm 2은 확장된 에지 제거 비용을 이용하여 다수의 메쉬에 대한 동시 단순화 알고리즘을 나타낸다.

```
// Extended edge collapse information
struct ExtCollapseInfo{
    int i;           // index of edge  $e_i$ 
    int n;           // number of meshes  $M^1, M^2, \dots, M^n$ 
    float c[];        //  $c[j]$ : collapse cost of  $e_i$  in  $M^j$ 
    float w[];        //  $w[j]$ : weight for  $c[j]$ 
    float tot_c;      // weighted cost of  $e_i$ 
    position v[];     //  $v[j]$ : collapse position in  $M^j$ 
};
```

Figure 6: Data structure for extended edge collapse information.

5. 실험 결과

본 논문에서 제안한 다수의 메쉬에 대한 동시 단순화 기법은 C++ 언어를 사용하여 Intel i7-7700K 4.2 GHz CPU, 8GB의 메인 메모리와 NVIDIA GeForce GTX 1060 그래픽 카드가 설치된 PC에서 구현되었다. 삼각 메쉬를 표현하기 위해 Kettner [16]가 제안한 하프에지(half edge) 자료구조를 활용하였다. 삼각 메쉬에 대한 하프에지 자료구조는 메쉬의 기하학적 연결 정보를 실시간으로 빠르게 얻을 수 있기 때문에 제안된 기법의 계산 성능 향상에 기여할 수 있다.

본 논문에서는 제안된 알고리즘의 유효성을 실험하기 위해 다양한 표정을 가진 얼굴 모델을 사용하였다. 각각의 모델은 정점이 4573개, 폴리곤이 9108개로 모두 동일했고 메쉬의 연결구조 또한 동일하다. Figure 7(a)는 2개의 모델 사이에 블렌드쉐이프를 적용

Algorithm 2: Simultaneous simplification of multiple triangle meshes.

Input: $\mathcal{M} = \{M^1, \dots, M^N\}$: a set of input meshes,
 $vnum$: target number of vertices

Output: \mathcal{M}' : a set of simplified meshes

for each vertex v_i do

for each mesh $M^j \in \mathcal{M}$ do

 compute Q_i^j ;

end

end

for each edge e_i do

for each mesh M^j do

 compute collapse position $v_i'^j$ and cost C_i^j ;

end

 compute standard deviation σ_i and weights w_j ;

 compute weighted cost C_i ;

 push $(e_i, v_i'^j, C_i)$ into min-heap;

end

while # of vertices $> vnum$ do

 pop $(e_i, v_i'^j, C_i)$ from min-heap;

for each mesh M^j do

 collapse e_i into $v_i'^j$;

for each edge e_k incident to $v_i'^j$ do

 compute collapse position $v_k'^j$ and cost C_k^j ;

end

end

for each modified edge e_i do

 compute standard deviation σ_i and weights w_j ;

 compute weighted cost C_i ;

 push $(e_i, v_i'^j, C_i)$ into min-heap;

end

end

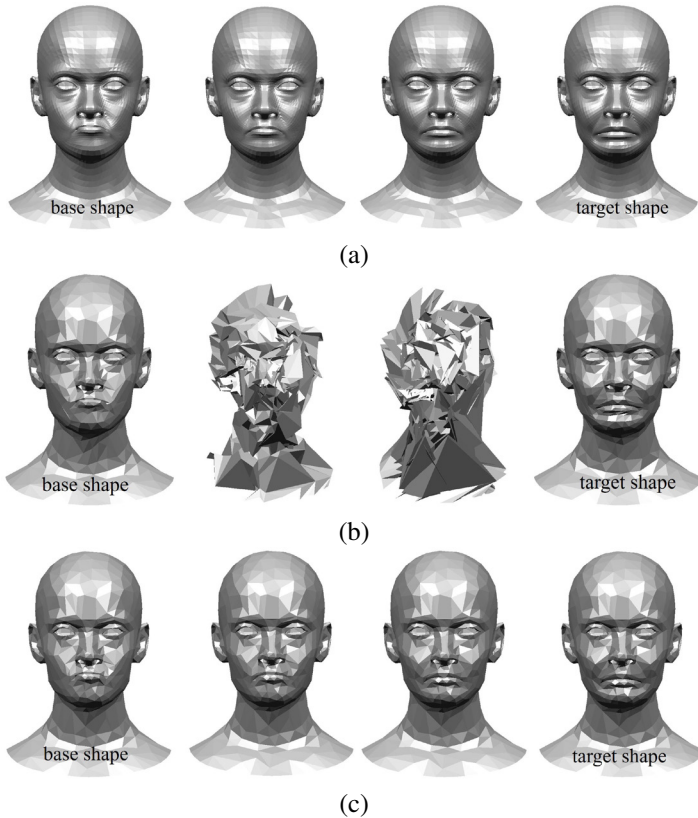


Figure 7: The process of creating blend shapes from a base shape and a target shape: (a) original models (4573 vertices), (b) separately simplified models (914 vertices), (c) simultaneously simplified models (914 vertices).

하여 애니메이션이 진행되는 과정을 보여준다. Garland와 Heckbert [10]의 알고리즘(Algorithm 1)을 베이스쉐입과 타겟쉐입 각각에 대해 적용하면 모델간에 일대일 대응관계가 더이상 성립하지 않아 선형보간이 제대로 적용되지 않는 반면(Figure 7(b)), 제시된 동시 단순화 알고리즘(Algorithm 2)을 적용하면 단순화 후에도 모델간에 선형보간이 잘 이루어진다(Figure 7(c)). Figure 8은 제시된 알고리즘의 확장성과 안정성을 보이기 위해서 서로 다른 3개의 얼굴 모델에 대하여 동일한 실험을 수행한 결과이다. Table 1은 메쉬의 수를 늘려가면서 동시 단순화에 소요되는 시간을 측정한 결과이고, Figure 9은 Table 1의 결과를 가시화한 결과이다. 실험 결과에서 보듯이 메쉬의 수에 따라 동시 단순화를 수행하는데 소요되는 시간은 선형적으로 증가하는 것을 볼 수 있다.

6. 결론

본 논문에서는 다수의 삼각 메쉬에 대하여 동시 단순화를 수행하는 알고리즘을 제안하였다. 각각의 메쉬에 대하여 에지 제거 비용을 분석하여 모든 메쉬의 형상을 고려하여 확장된 에지 제거 정보를 설계하였으며, 이를 기반으로 모든 메쉬에 공통적으로 적용될 수 있는 최적화된 에지 제거 순서를 결정하였다. 동일한

정점의 수와 연결 정보를 갖는 다양한 얼굴 메쉬에 대하여 동시 단순화 알고리즘을 적용하는 실험을 수행한 결과, 결과 메쉬들은 모두 동일한 정점의 수와 연결 정보를 유지하면서 사용자가 지정한 수준으로 단순화되는 결과를 보였다. 메쉬의 수를 최대 30개까지 변화시키는 실험을 통해 제시된 기법의 안정성과 유효성을 입증하는 실험을 수행하였고, 이를 통해 동시 단순화 기법은 메쉬에 수에 따라 선형적으로 증가하는 시간 복잡도를 갖는다는 것을 실험적으로 보였다.

제시된 방법은 메쉬의 기하 정보만 고려하였기 때문에 텍스처가 입혀진 모델에 대해서는 단순화 결과가 어색하다. 또한 얼굴 모델 텍스처의 경우에는 눈, 머리카락, 피부로 나뉘기 때문에 텍스처 경계부분에서 알고리즘이 잘 적용되지 않는다. 향후 연구에서는 텍스처 좌표 속성을 포함하고 있는 모델에 대해서 텍스처 경계부분이 잘 유지되도록 동시 단순화 기법을 확장할 계획이다.

감사의 글

이 성과는 2019년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구(No. NRF-2018R1D1A1B07048036)와, 2018년 과학기술정보통신부 방송통신발전기금으로 한국전파진흥협회의 지원을 받아 수행된 연구임

References

- [1] 신유성, “브랜드쉐입을 이용한 캐릭터제작 파이프라인 구축에 관한 연구,” *커뮤니케이션 디자인학연구*, vol. 49, no. 2, pp. 86–95, 2014.
- [2] J. P. Lewis, K. Anjyo, T. Rhee, M. Zhang, F. Pighin, and Z. Deng, “Practice and Theory of Blendshape Facial Models,” in *Eurographics 2014 - State of the Art Reports*. The Eurographics Association, 2014.
- [3] F. I. Parke, “Computer generated animation of faces,” in *Proceedings of the ACM Annual Conference - Volume 1*, 1972, pp. 451–457.
- [4] J. Lewis, J. Mooser, Z. Deng, and U. Neumann, *A User Interface Technique for Controlling Blendshape Interference*, 2007, pp. 132–144.
- [5] J. Seo, G. Irving, J. Lewis, and J. Noh, “Compression and direct manipulation of complex blendshape models,” in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6, 2011, p. 164.
- [6] T. Costigan, A. Gerdeman, E. Carrigan, and R. McDonnell, “Improving blendshape performance for crowds with GPU and GPGPU techniques,” in *Proceedings of the 9th International Conference on Motion in Games*, 2016, pp. 73–78.

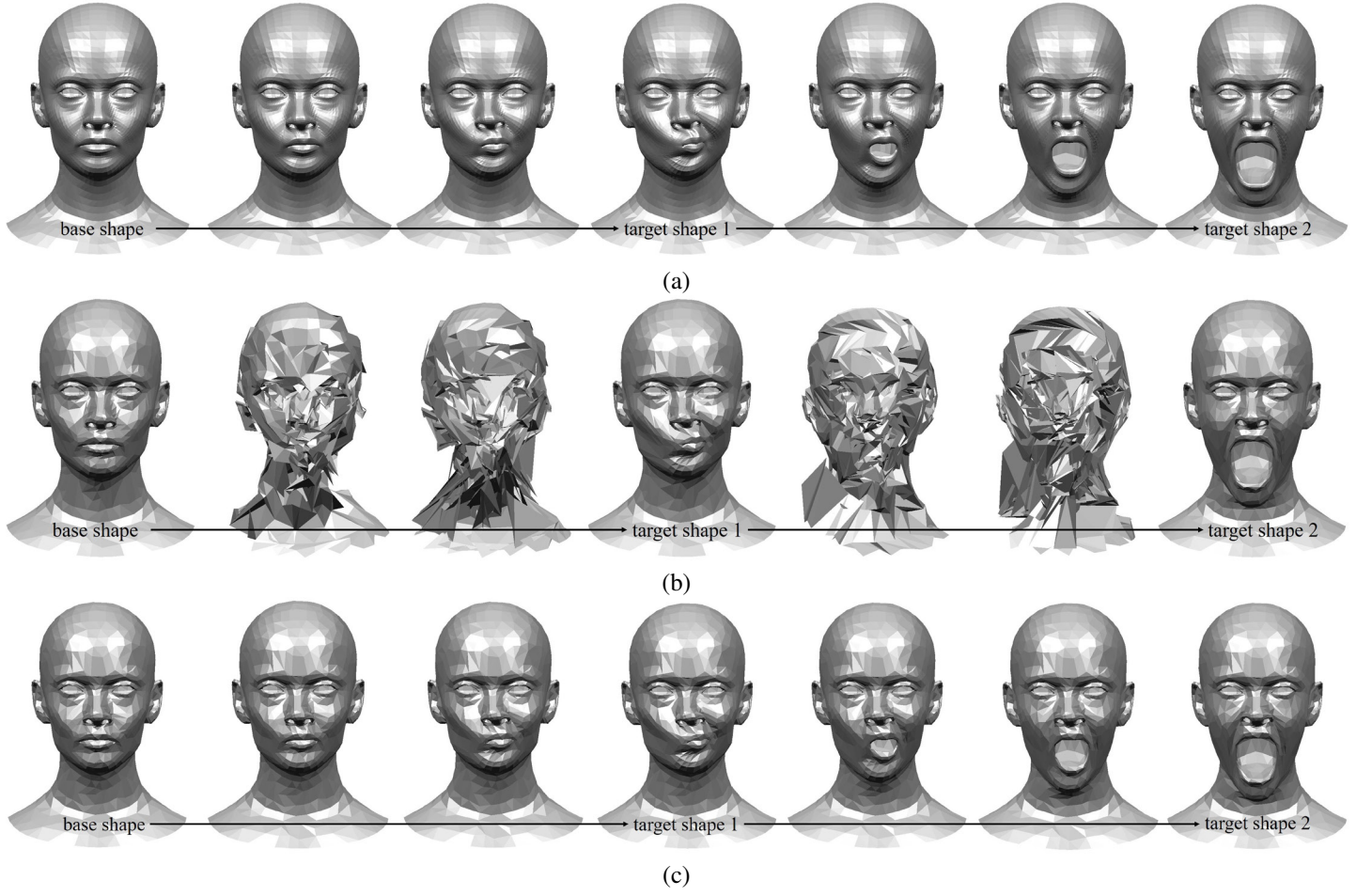


Figure 8: The process of creating blend shapes from a base shape and a target shape: (a) original models (4573 vertices), (b) separately simplified models (1371 vertices), (c) simultaneously simplified models (1371 vertices).

Table 1: Computational time.

# meshes	Time (ms)		
	Simplification rate(50%)	Simplification rate(25%)	Simplification rate(10%)
2	43	80	83
4	86	136	154
6	163	196	234
8	189	264	313
10	218	329	393
12	276	409	482
14	305	463	552
16	356	523	625
18	397	583	693
20	446	676	786
22	501	737	876
24	541	841	949
26	582	864	1025
28	632	924	1100
30	679	1006	1195

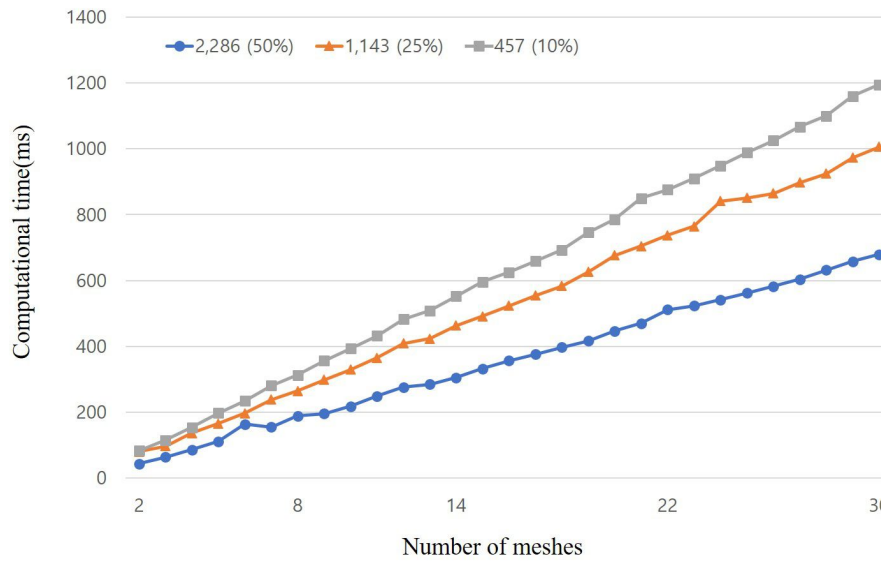


Figure 9: Computational time of simultaneous simplification for different number of meshes.

- [7] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen, "Decimation of triangle meshes," *SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 65–70, 1992.
- [8] G. Turk, "Re-tiling polygonal surfaces," *SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 55–64, 1992.
- [9] A. D. Kalvin and R. H. Taylor, "Superfaces: Polygonal mesh simplification with bounded error," *IEEE Computer Graphics and Applications*, vol. 16, no. 3, pp. 64–77, 1996.
- [10] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '97, 1997, pp. 209–216.
- [11] A. Guéziec, *Surface simplification inside a tolerance volume*. IBM TJ Watson Research Center, 1996.
- [12] P. Cignoni, C. Montani, and R. Scopigno, "A comparison of mesh simplification algorithms," *Computers & Graphics*, vol. 22, no. 1, pp. 37–54, 1998.
- [13] M. Pauly, M. Gross, and L. P. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Proceedings of the conference on Visualization'02*, 2002, pp. 163–170.
- [14] S.-J. Kim, S.-K. Kim, and C.-H. Kim, "Discrete differential error metric for surface simplification," in *In Proc. of 10th Pacific Conference on Computer Graphics and Applications*, 2002, pp. 276–283.
- [15] R. Yi, Y.-J. Liu, and Y. He, "Delaunay mesh simplification with differential evolution," in *SIGGRAPH Asia 2018 Technical Papers*. ACM, 2018, p. 263.
- [16] L. Kettner, "Using generic programming for designing a data structure for polyhedral surfaces," *Computational Geometry*, vol. 13, no. 1, pp. 65–90, 1999.
- [17] M. Garland and P. S. Heckbert, "Simplifying surfaces with color and texture using quadric error metrics," in *In Proc. of Visualization'98*, 1998, pp. 263–269.

〈 저 자 소 개 〉



박 정 호

- 2019 동국대학교 멀티미디어공학과 학사
- 2019 ~ 현재 동국대학교 멀티미디어공학전공 석사과정
- 관심분야 : 기하 모델링, 가상현실, 컴퓨터 그래픽스
- <https://orcid.org/0000-0002-6708-6692>



김 중 용

- 2014 청운대학교 컴퓨터학과 학사
- 2016 동국대학교 멀티미디어학과 석사
- 2016 ~ 현재 동국대학교 멀티미디어학과 박사과정
- 관심분야 : 혼합현실, 가상현실, 컴퓨터 그래픽스
- <https://orcid.org/0000-0001-8062-3678>



송 중 훈

- 2018 동국대학교 멀티미디어공학과 학사
- 2018 ~ 현재 동국대학교 멀티미디어학과 석사과정
- 관심분야 : 혼합현실, 가상현실, 실시간 렌더링
- <https://orcid.org/0000-0003-1384-0466>



박 상 훈

- 1993 서강대학교 수학과 학사
- 1995 서강대학교 컴퓨터학과 석사
- 2000 서강대학교 컴퓨터학과 박사
- 2002 ~ 2005 대구카톨릭대학교 컴퓨터정보통신공학부조교수
- 2001 University of California, Davis 방문 연구원
- 2005 ~ 현재 동국대학교 멀티미디어학과 교수
- 관심분야: 실시간 렌더링, 사실적 렌더링, 과학적 가시화, 고성능 컴퓨팅 등
- <https://orcid.org/0000-0001-5383-7005>



윤 승 현

- 2001 한양대학교 수학과 학사
- 2007 서울대학교 컴퓨터공학과 박사
- 2007 ~ 현재 동국대학교 멀티미디어공학과 조교수/부교수/교수
- 관심분야 : 컴퓨터그래픽스, 기하모델링, 가상/증강/혼합현실
- <https://orcid.org/0000-0002-0015-8305>