

일반논문-08-13-2-05

운동 특성 벡터에 기반한 메쉬 에디팅 기법

이순영^{a)}, 김창수^{b)}, 이상욱^{a)*}

Mesh Editing Using the Motion Feature Vectors

Soon-Young Lee^{a)}, Chang-Su Kim^{b)} and Sang-Uk Lee^{a)*}

요 약

본 논문에서는 두 개의 3차원 메쉬 프레임에서 추출한 운동 성질을 이용하여 형태 특징을 반영하는 3차원 메쉬 에디팅 알고리즘을 제안하였다. 제안하는 알고리즘은 형태적 특징을 유지하는 결과를 얻기 위해서 두 개의 메쉬 프레임으로부터 국부 영역의 표면 특징을 반영하는 운동 특성 벡터를 이용한다. 에디팅 과정에서 사용자는 임의의 꼭지점을 조작 꼭지점(anchor vertex)으로 선택하여 자유롭게 이동시킨다. 조작 꼭지점에는 사용자의 조작에 따라 강제된 운동이 부여되고, 이 운동 특성을 유지하도록 이웃 꼭지점들의 위치가 정해진다. 이 과정을 통해 샘플 메쉬의 특징을 보존하는 에디팅 결과를 얻는다. 모의 실험을 통하여 제안하는 알고리즘이 사용자가 의도한 조작을 충분히 반영하는 동시에 샘플 메쉬 상에서 나타나는 운동성을 보존하는 결과를 도출함을 확인하였다.

Abstract

In this paper, we proposed a new mesh editing algorithm based on the motion between two sample meshes. First, the motion vectors are defined as the derivation vector of the corresponding vertices on the sample meshes. Then, the motion feature vectors are extracted between the motion vectors. The motion feature vectors represent the similarity of the vertex motion in a local mesh surface. When a mesh structure is forced by an external motion of anchor vertices, the deformed mesh geometry is obtained by minimizing the cost function with preserving the motion feature vectors. Simulation results demonstrated that the proposed algorithm yields visually pleasing editing results.

Keyword : mesh editing, mesh deformation, motion feature vector

1. 서 론

최근 인터넷과 하드웨어가 크게 발달함에 따라 멀티미디어 자료를 일상에서 쉽게 접할 수 있게 되었다. 이 중 2차원

영상 정보는 시점이 자유롭지 못하고 현실감이 부족한 한계를 지닌다. 이러한 2차원 영상 정보의 단점을 보완한 3차원 영상 정보는 현재 영화, 게임, 애니메이션 등에서 활용되고 있으며 응용 분야도 더욱 확대될 것으로 예상된다.

3차원 영상 정보는 여러 방법으로 표현되지만 특히 렌더링이 빠르고 비교적 작은 크기를 갖는 삼각형 메쉬 데이터 형식이 널리 사용된다. 3차원 삼각형 메쉬 데이터는 만드는 방법에 따라 취득 데이터와 컴퓨터 그래픽 데이터로 분류할 수 있다. 취득 데이터는 물체의 표면을 레이저 스캐너로

a) 서울대학교 전기컴퓨터공학부, 뉴미디어통신공동연구소
Signal Processing Lab., School of Electrical Engineering and INMC,
Seoul National University, Seoul, Korea
b) 고려대학교 전기전자전파공학부*
School of Electrical Engineering, Korea University, Seoul, Korea*
* 교신저자 : 이상욱(sanguk@ipl.snu.ac.kr)

샘플링한 후 이를 공간상의 꼭지점으로 표현하여 3차원 영상 정보로 나타낸 것이다. 그래픽 데이터는 컴퓨터 등을 이용하여 가상의 공간에 점, 선, 면 등의 형태소(primitive)를 배치하고 이를 편집하여 3차원 영상 정보를 얻는 것이다. 이러한 두 방법은 모두 강도 높은 수작업이 요구되어 시간이 오래 걸리는 단점을 지닌다.

따라서 많은 양의 3차원 영상 데이터를 위와 같은 방법으로 직접 만드는 것은 상당히 비효율적이다. 이에 따라 이전에 생산한 3차원 영상 데이터를 재사용하여 새로운 3차원 영상 정보로 만드는 가공(manipulation) 기법이 요구된다. 이러한 가공 기법 중 하나가 메쉬 에디팅 기법이다. 메쉬 에디팅은 3차원 데이터의 일부 혹은 전체에 외부 운동을 가하였을 때 이를 반영하면서 전체의 모양을 자연스럽게 결정하는 기법을 의미한다. 최근에는 사용자가 원하는 부분에 임의의 변형을 가하는 자유 형태 변형(free form deformation) 에디팅 기법에 대한 연구가 활발하게 진행되고 있다^[1-8].

초기 에디팅 알고리즘으로 Sederberg은 모델 주위를 둘러싼 제어점(control points)을 이용해 외부 직육면체에 변형을 가함으로써 내부의 3차원 모델을 변형시키는 알고리즘을 제안하였다^[3]. 자유 형태 변형 기법으로는 해상도가 다른 여러 데이터를 이용하여 큰 영역에서의 변형과 함께 좁은 영역에서의 변형을 고려하는 다중 해상도(multiresolution) 에디팅 알고리즘이 제안되었다^[4,5]. 이밖에도 삼각형 메쉬 데이터가 가지고 있는 정보를 에디팅에 효율적이도록 가공한 매개변수로 변형을 수행하는 내부 매개변수(intrinsic parameter) 에디팅 알고리즘이 제안되었다^[1,6,7]. 이러한 기존 알고리즘들은 대부분 에디팅 과정에서 단일 메쉬 데이터만을 사용하기 때문에 모델의 형태적 특징을 반영하기에는 한계가 있다. 이를 보완하기 위해 Sumner는 여러 개의 메쉬 데이터의 대응하는 삼각형에서 어파인 변형 행렬(affine transform matrix)을 추출하고 이를 조합하여 만든 특징 공간(feature space)에서의 변형을 통한 에디팅 방법을 제안하였다^[8]. 하지만 이 알고리즘은 형태를 보존하는 에디팅 결과를 도출하였지만 움직임에 대한 고찰이 부족하다.

본 논문에서는 3차원 메쉬 데이터의 운동 특성에 기반한 새로운 에디팅 알고리즘을 제안한다. 제안하는 알고리즘은 사용자가 임의로 모델의 일부 꼭지점을 조작 꼭지점(anchor

vertex)으로 설정하고 이 꼭지점들을 공간에 자유롭게 위치 시킴에 따라 모델의 형태를 변화시키는 자유 형태 변형(free-form deformation)의 맥락에서 이루어진다. 먼저 주어진 3차원 메쉬 데이터에서 운동 성질을 추출한 각 꼭지점 사이의 운동벡터(vertex-wise motion vector)를 이용하여 운동 성질을 표현한다. 그리고 공간적으로 인접한 꼭지점에서의 운동 벡터들이 가지는 관련성을 관찰하여 국부 영역에서 운동 벡터들이 이루는 비율을 형태를 결정하는 운동 특성(motion feature) 벡터로 매개변수화한다. 그리고 사용자가 인가하는 조건을 만족시키면서 운동 특성 벡터를 유지하는 3차원 메쉬의 꼭지점들을 평균 제곱 오류 최소화(mean square error minimization)를 이용하여 얻는다.

본 논문의 구성은 다음과 같다. 2장에서 메쉬의 표면의 특징을 나타내는 운동 특성 벡터를 정의하고 이를 이용하여 새로운 에디팅 알고리즘을 제안한다. 3장에서 모의 실험을 통하여 제안하는 알고리즘과 기존 알고리즘을 비교한다. 마지막으로 4장에서 결론과 향후 과제를 제시한다.

II. 제안하는 알고리즘

1. 꼭지점 사이의 운동 벡터

영상 데이터에서의 운동은 시간에 따라 대응(corresponding)하는 목표 객체의 위치 차이로 표현된다. 이를 나타낸 운동 벡터는 시간적으로 상관 관계를 가지는 2차원 시퀀스 데이터를 분석하는데 널리 사용되고 있다^[9-11]. 시퀀스로 구성된 3차원 영상 정보에서도 이와 같은 방법으로 객체의 운동을 표현한다. 특히 3차원 영상 정보에서의 운동 벡터는 공간 상의 움직임을 나타내야 하므로 독립적인 3차원 벡터로 표현한다.

본 논문에서는 두 샘플 메쉬 프레임 간의 운동성을 표현하기 위하여 꼭지점간의 운동벡터를 이용한다. N 개의 꼭지점과 동일한 연결정보를 가지는 샘플 메쉬를 기준 메쉬 M_r 과, 움직임이 나타난 목적 메쉬 M_t 로 정의한다. 각 메쉬에서 대응하는 i 번째 꼭지점을 각각 $\mathbf{v}_r(i) \in M_r, \mathbf{v}_t(i) \in M_t$ 라고 할 때 꼭지점 사이의 운동 벡터는 $\mathbf{mv}_{rt}(i) = \mathbf{v}_t(i) - \mathbf{v}_r(i)$ 로

정의된다. $mv_{rt}(i)$ 는 각 꼭지점에서의 운동 성분의 크기와 방향을 나타낸다. 이 벡터는 제안하는 알고리즘에서 삼각형 메쉬의 데이터의 지역 표면(local surface)을 특징짓는데 사용된다.

2. 표면 형태에 기반한 운동 특성

3차원 삼각형 메쉬는 물체의 표면을 표현하는 형태소로써 삼각형을 이용한다. 한 쌍의 꼭지점이 한 삼각형을 구성하는 꼭지점일 때 이를 이웃 꼭지점(neighbor vertex)이라 정의한다. 3차원 공간에서의 운동은 x, y, z의 세 성분으로 나타난다. 그리고 각 성분은 x, y, z 정규 직교 좌표축에 대한 독립적인 운동을 의미하기 때문에 각 성분을 분리해도 운동 벡터가 가지는 특징을 보존한다. 따라서 본 논문에서는 한 꼭지점에서의 운동 벡터와 그 이웃 꼭지점에서의 운동 벡터들의 관계를 축성분별로 분리하여 도출한다. 이렇게 샘플 메쉬상으로부터 얻은 꼭지점 사이의 운동 벡터 mv_{rt} 를 x, y, z 축성분으로 분리한 벡터를 성분별 벡터(component-wise vector)라고 정의한다. 그림 1은 0번째 꼭지점을 대상 꼭지점으로 했을 때 이웃 꼭지점에서 꼭지점

사이의 운동 벡터를 성분별 벡터로 나타낸 것이다. 빨간색으로 표시한 성분은 대상 꼭지점의 운동을, 파란색으로 표시한 성분은 이웃 꼭지점에서의 운동을 표현한다. 각 성분별로 표시한 그래프는 운동 벡터의 크기를 나타내고 축의 인덱스 i 는 i 번째 꼭지점을 의미한다.

지역 표면에서 운동 벡터들이 가지는 연관성을 이용하여 한 꼭지점에서의 성분별 운동 벡터와 이웃 꼭지점에 있는 성분별 운동 벡터 사이의 관계를 찾는다. 그림 2(a)와 같이 대상 꼭지점과 그 이웃의 운동 벡터들이 정의되었을 때 z-성분 운동 벡터는 그림 2(b)와 같이 표현된다. 이 때 빨간색으로 표시한 벡터는 대상 꼭지점에서의 운동 벡터이고 파란색으로 표시한 벡터는 이웃 꼭지점에서 운동 벡터이다. 지역 표면에서는 운동 성질이 유사하기 때문에 대상 꼭지점의 운동은 이웃 꼭지점에서의 운동과 비례하는 관계로 가정한다. 따라서 제안하는 알고리즘은 대상 꼭지점에서의 운동 벡터와 그 이웃 꼭지점에서의 운동 벡터가 이루는 비율로써 운동 특성을 표현한다.

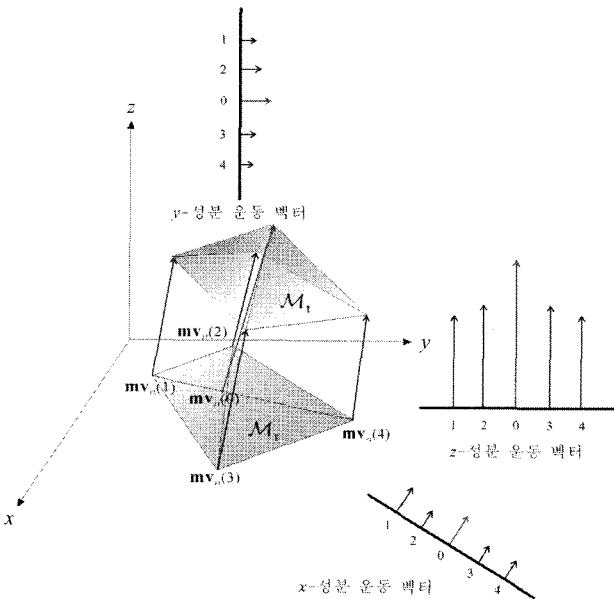


그림 1. 성분별 운동벡터.
Fig 1. Component-wise motion vectors.

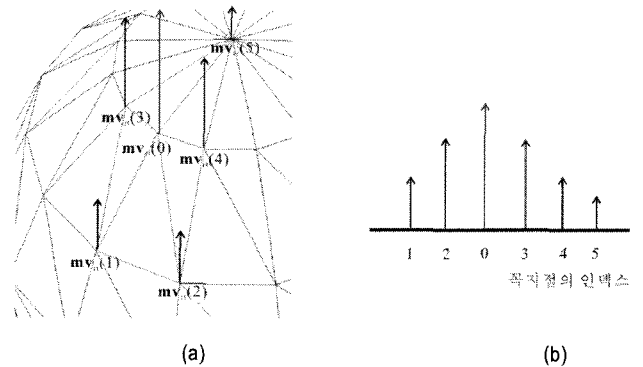


그림 2. 국부 영역에서의 운동 벡터 사이의 관계.
Fig 2. Motion vectors in local region around $v_r(0)$.

먼저 기준 메쉬 M_r 의 i 번째 꼭지점 $v_r(i)$ 를 대상 꼭지점으로 선택하고 $v_r(i)$ 와 일환이웃(1-ring neighbor)인 꼭지점들의 집합을 $N_{v_r(i)} = \{v_r(j) | j=0,1, \dots, K_i-1\}$ 라 정의한다. 그리고 $mv_{rt}(i)$ 와 $v_r(j) \in N_{v_r(i)}$ 의 운동 벡터 $mv_{rt}(j)$ 사이에서 다음과 같은 선형 관계식을 도출한다.

$$mv_{rt}(i) = f(i) \odot \frac{1}{K_i} \sum_{j=0}^{K_i-1} mv_{rt}(j). \quad (1)$$

여기서 $\mathbf{f}(i)$ 는 $(f_x(i), f_y(i), f_z(i))^T$ 인 3차원 계수 벡터이고, $(p_x, p_y, p_z) \odot (q_x, q_y, q_z)$ 는 $(p_x q_x, p_y q_y, p_z q_z)$ 인 벡터 연산을 나타낸다. 계수 벡터 $\mathbf{f}(i)$ 는 x, y, z 각 좌표 성분별로 대상 꼭지점의 운동 벡터와 이웃 꼭지점들의 운동 벡터들의 평균과의 비율을 의미한다.

계수 벡터 $\mathbf{f}(i)$ 는 i 번째 대상 꼭지점 근처의 지역 표면의 물리적 성질을 나타낸다. 지역 표면에서 꼭지점의 운동은 평행 운동으로 근사할 수 있다. 만약 대상 꼭지점 근처가 딱딱한(rigid) 영역이기 때문에 M_r 과 M_t 사이에서 모양이 쉽게 변하지 않는다면 대상 꼭지점과 그 이웃 꼭지점의 운동 벡터는 매우 유사하다. 그림 3(a)는 딱딱한 영역에서 운동 벡터의 z성분만을 나타낸 것이다. 이와 같은 경우에는 운동 특성 $f_z(i)$ 는 거의 1에 가까운 값을 가진다. 이와는 반대로 그림 3(b)와 같이 대상 꼭지점 근처가 부드러운(soft) 영역으로 쉽게 변형이 일어난다면 대상 꼭지점에서의 운동 벡터는 이웃 꼭지점의 운동 벡터와 큰 차이를 보일 것이며 $f_z(i)$ 은 1과 큰 차이를 보인다. 따라서 $\mathbf{f}(i)$ 는 대상 꼭지점 근처의 표면 특성을 나타내는 운동 특성 벡터(motion feature vector)의 의미를 갖는다.

3. 사용자 상호 작용에 기반한 에디팅

제안하는 알고리즘에서는 3차원 메쉬의 변형 결과가 사용자의 변형 의도를 그대로 반영한다. 사용자는 주어진 기준 메쉬 M_r 의 임의의 꼭지점을 조작 꼭지점(anchor vertex)으로 선택하여 새로운 좌표로 자유롭게 이동시킨다. 에디팅 결과는 이러한 사용자의 변형 의도를 반영하는 동시에 3차원 메쉬 모

델의 형태적 특징을 유지하여야 한다. 따라서 제안하는 알고리즘에서는 움직임 특징을 보존하면서 메쉬 전체의 좌표를 재구성함으로써 움직임이 자연스러운 에디팅 결과를 얻는다.

제안하는 알고리즘에서는 사용자가 3차원 데이터의 꼭지점을 선택하여 임의로 움직임으로써 원하는 변형을 유도한다. 먼저 사용자가 선택한 L 개의 조작 꼭지점의 집합을 $S = \{v_r(l) | l=0, 1, \dots, L-1\}$ 로 정의한다. 그리고 이 조작 꼭지점 $v_r(l) \in S$ 을 새로운 위치 $\hat{v}_r(l)$ 으로 이동시킨다. 이 때 $v_r(l)$ 에 새롭게 정의되는 운동벡터를 $\hat{m}\mathbf{v}(l) = \hat{v}_r(l) - v_r(l)$ 이라고 나타낸다. 에디팅은 조작 꼭지점에 가해진 움직임 $\hat{m}\mathbf{v}(l)$ 에 의하여 운동 특성 벡터가 변하지 않도록 주변 꼭지점들의 위치를 정함으로써 이루어진다. 변형이 이루어진 3차원 삼각형 메쉬를 라고 하면 에디팅 결과는 다음과 같은 식으로 얻어진다.

$$M_d = \arg \min_{\mathbf{v}_d(l)} \left[\sum_{i=0}^{N-1} \|\mathbf{m}\mathbf{v}_{rd}(i) - \mathbf{f}(i)\|^2 \odot \frac{1}{K} \sum_{v_d(j) \in N_d(i)} \|\mathbf{m}\mathbf{v}_{rd}(j)\|^2 \right], \quad (2)$$

subject to $\mathbf{m}\mathbf{v}_{rd}(l) = \hat{m}\mathbf{v}(l)$ for each $v_r(l) \in S$.

여기서 $\mathbf{m}\mathbf{v}_{rd}(i) = v_r(i) - v_d(i)$ 이고 $\|\cdot\|$ 는 벡터 크기(vector norm)를 나타낸다. 식 (2)를 통해 주어진 조작 꼭지점들에서 운동 특성 벡터가 유지되는 변형된 꼭지점의 위치 $v_d(i)$ 를 얻고 전체 결과 메쉬 M_d 를 도출한다.

이 과정은 다음과 같은 행렬 관계식으로 표현이 가능하다.

$$\arg \min_{\mathbf{m}} \|\mathbf{L}\mathbf{m}\|^2, \quad (3)$$

subject to $\mathbf{m}\mathbf{v}_{rd}(l) = \hat{m}\mathbf{v}(l)$ for each $v_r(m) \in S$.

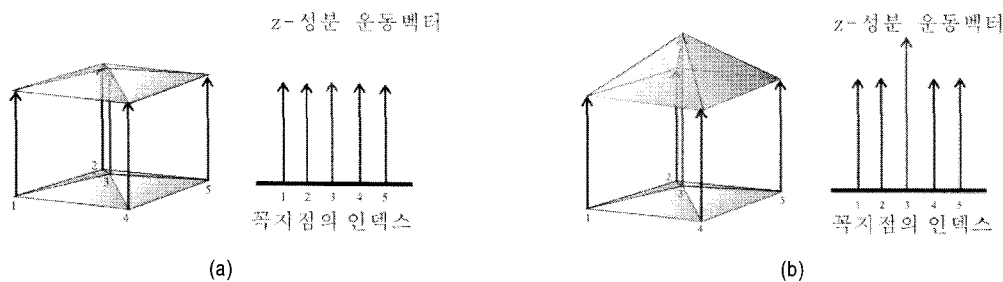


그림 3. 운동 특성 벡터의 물리적 의미: (a) 딱딱한 영역, (b) 부드러운 영역.
Fig 3. Physical meaning of feature vectors: (a) rigid region and (b) elastic region.

여기서 \mathbf{m} 은 $\mathbf{mv}_{rd}(i)$ 을 각 성분별로 나열한 것으로 다음과 같이 정의된다.

$$\mathbf{m} = (mv_{rd}^{(x)}(0), \dots, mv_{rd}^{(y)}(0), \dots, mv_{rd}^{(z)}(0), \dots, mv_{rd}^{(z)}(N-1))^T \quad (4)$$

여기서 $\mathbf{mv}_{rd}(i) = (mv_{rd}^{(x)}(i), mv_{rd}^{(y)}(i), mv_{rd}^{(z)}(i))^T$ 이다. 그리고 행렬 \mathbf{L} 은 $3N \times 3N$ 크기로 운동 특성 벡터를 이용하여 다음과 같이 구성된다.

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}^{(x)} & 0 & 0 \\ 0 & \mathbf{L}^{(y)} & 0 \\ 0 & 0 & \mathbf{L}^{(z)} \end{pmatrix} \quad (5)$$

여기서 $\mathbf{L}^{(x)}$ 는 아래와 같은 $N \times N$ 행렬이고 윗첨자 (x), (y), (z)는 운동 특성 벡터의 x, y, z 성분임을 나타낸다.

$$\mathbf{L}_{ij}^{(x)} = \begin{cases} 1 & i = j \\ -f_x(i) & i \neq j, j \in N_{v_r(i)} \\ 0 & otherwise \end{cases} \quad (6)$$

식 (3)과 같이 선형 관계로 표현된 목적 함수의 최적해 \mathbf{m} 은 의사 역행렬(pseudo-inverse matrix)나 가우스-뉴튼 방법(Gauss-Newton method)을 이용하여 구할 수 있다. 이 최적해 \mathbf{m} 은 기준 메쉬 M_r 과 에디팅 결과 메쉬 M_d 에서 꼭지점 사이의 움직임에 의미한다. 따라서 M_r 의 각 꼭지점 좌표에 이동량 $\mathbf{mv}_{rd}(i)$ 을 더하여서 사용자의 변형 의도가 반영된 결과 메쉬 M_d 를 얻는다.

III. 실험결과

제안하는 알고리즘의 성능을 평가하기 위해 3차원 삼각형 메쉬로 표현된 여러 데이터를 이용하여 에디팅을 수행하였다. 제안하는 알고리즘의 에디팅 결과를 기존의 Alex가 제안하였던 단일 메쉬를 이용한 알고리즘^[1]과 비교한다. 실험에 사용된 데이터는 3차원 메쉬로 이루어진 시퀀스 데이터에서 두 프레임을 추출한 한 것으로 각 샘플 메쉬 쌍은 동일한 연결정보를 가지고 동일한 꼭지점 인덱스를 가지고 있다. 알고리즘은 C++와 OpenGL을 이용하여 구현하였고 동작 환경은 펜티엄 3.0기가 헤르츠 중앙 처리장치와 3기가 바이트의 메모리가 장착된 PC이다.

그림 4는 "Stick" 데이터를 이용하여 결과를 비교한 것이다. 분홍색 메쉬는 M_r , 회색 메쉬는 M_t , 노란색 메쉬는 제안하는 알고리즘의 결과, 붉은색 메쉬는 차분 좌표를 이용한 알고리즘의 결과를 나타낸다. 그림 4(a)와 그림 4(b)는 M_r 의 가운데 높이에 위치한 꼭지점을 오른쪽으로 이동시킨 결과이다. 제안하는 알고리즘의 결과를 나타낸 그림 4(a)는 두 메쉬 사이의 휘어지는 물리적 성질을 반영한 반면 단일 메쉬만을 사용한 그림 4(b)에서는 그 특징이 제대로 나타나지 않은 변형 결과를 보인다. 그림 4(c)와 그림 4(d)는 M_r 의 가장 위의 한 점을 아래로 이동시킨 결과이다. 마찬가지로 제안하는 알고리즘은 휘어짐을 제대로 나타낸 반면에 단일 메쉬만을 사용하였을 경우엔 그 성질이 반영되지 않아 움직임이 자연스럽게 못한 결과를 얻는다.

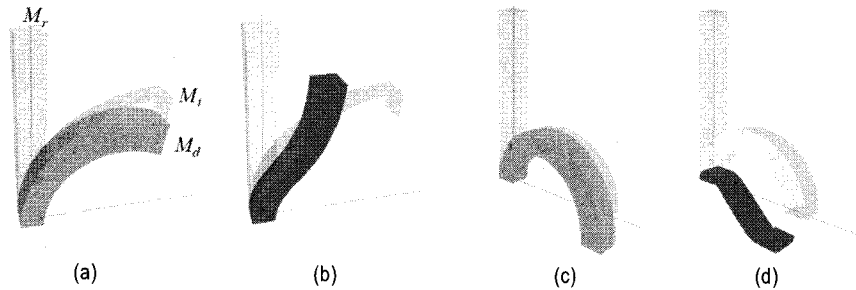


그림 4. "Stick" 모델의 에디팅 결과: (a)제안하는 알고리즘, (b)차분좌표를 이용한 알고리즘, (c)제안하는 알고리즘, (d)차분좌표를 이용한 알고리즘.
Fig 4. Editing result of the "Stick" model: (a) proposed algorithm, (b) differential coordinate algorithm, (c) proposed algorithm, (d) differential coordinate algorithm.

그림 5는 "Dance" 데이터를 이용한 결과이다. 이 실험에서는 같은 모델에 두가지의 운동 시나리오를 적용하였다. 첫 번째는 그림 5(a)와 그림 5(b)에서와 같이 사람 형태의 모델의 왼쪽 발을 조작 꼭지점으로 선택하여 이를 앞으로 당겨 위치시킨 것이고 다른 하나는 그림 5(c)와 그림 5(d)처럼 왼쪽 발을 뒤쪽으로 밀어서 위치시킨 것이다. 두 경우 모두 단일 메쉬를 이용한 차분 좌표 알고리즘의 결과에서는 그림 5(b)나 그림 5(d)처럼 두 데이터 사이에 움직임이 없는 오른쪽 다리에서 움직임이 발생하였고 몸체나 팔 부분에서의 형태가 일그러진 결과를 보인다. 하지만 제안하는 알고리즘의 결과는 조작 꼭지점의 위치를 반영하여 자연스러운 움직임을 나타내는 결과를 도출한다. 이와 같은 실험 결과는 제안하는 알고리즘이 관절을 가지는 모델에서 그 특징을 보존하면서 에디팅이 가능함을 보여준다.

그림 6은 "Bounce ball" 데이터에 제안하는 알고리즘을 적용한 결과이다. "Bounce ball" 샘플 메쉬쌍은 그림 6과 같이 공간적으로 떨어져 있어 전체적으로 큰 평행 운동 성분이 존재한다. 이 실험에서는 조작 꼭지점을 구의 상단부분의 한 점으로 선택하여 이 조작 꼭지점을 오른쪽으로 움직여 가면서 변형된 결과를 얻었다. 에디팅 결과 메쉬는 오른쪽으로 평행 운동 성분을 주게 됨에 따라서 점차 M_t 의 형태도 반영함을 보여준다.

그림 7는 "Dancer" 데이터를 이용하여 모델이 앞으로 걸어가는 모습을 제안 에디팅 기법으로 생성한 결과이다. 샘플 메쉬쌍으로 이용한 데이터는 그림에서 테두리로 표시한 데이터이고 임의로 조작 꼭지점을 선택하여 샘플 메쉬와 다른 8개의 데이터를 생성하였다. 샘플 메쉬 쌍에서 표현하고 있는 걸어가는 움직임이 효과적으로 반영된 결과 메쉬를 확인할 수 있다. 이 실험 결과는 제안하는 알고리즘을

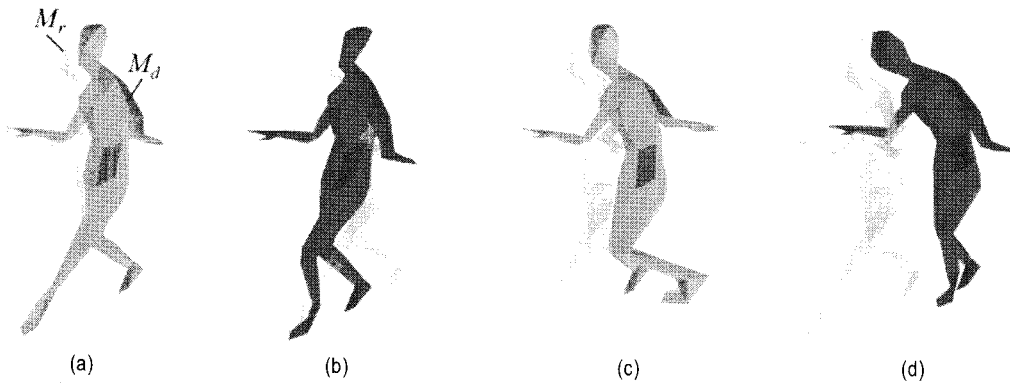


그림 5. "Dance" 모델의 에디팅 결과: (a)제안하는 알고리즘, (b)차분좌표를 이용한 알고리즘, (c)제안하는 알고리즘, (d)차분좌표를 이용한 알고리즘.
 Fig 5. Editing result of the "Dance" model: (a) proposed algorithm, (b) differential coordinate algorithm, (c) proposed algorithm, (d) differential coordinate algorithm.

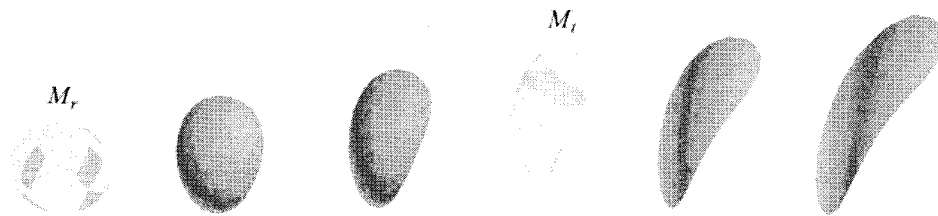


그림 6. "Bounce ball" 모델의 에디팅 결과.
 Fig 6. Editing result of "Bounce ball" model.

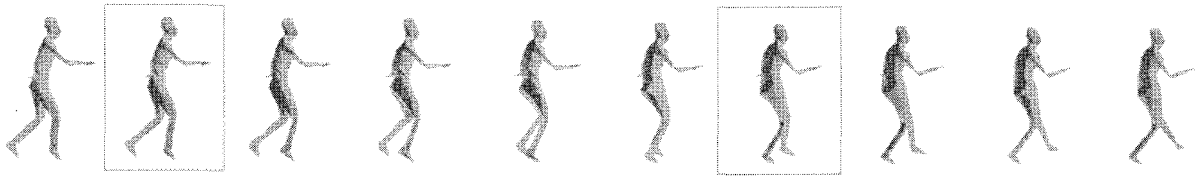


그림 7. "Dance" model을 이용한 시퀀스 생성.

Fig 7. Creation of a sequence data using two frames of "Dancer".

연속적인 시퀀스 데이터를 제작하는데 응용할 수 있음을 보여준다.

IV. 결론

본 논문에서는 운동 벡터를 이용하여 3차원 영상 정보에 에디팅하는 새로운 알고리즘을 제안하였다. 제안하는 알고리즘은 두 개의 메쉬 데이터에서 나타나는 운동성을 이용하여 메쉬 데이터가 가지는 특징을 추출하고 이를 보존하는 결과를 도출한다. 제안하는 알고리즘에서는 한 꼭지점과 인접한 이웃 꼭지점들에서의 꼭지점 사이의 운동 벡터 사이의 관계를 이용하여 주어진 샘플 메쉬 쌍의 형태적 특징을 매개변수화 한다. 한 꼭지점에서의 운동 벡터와 그 이웃에서의 꼭지점들의 운동 벡터들이 이루는 비율을 각 축 성분별로 표현하고 이를 샘플 메쉬의 운동 특성 벡터로 정의하였다. 에디팅 과정은 사용자 상호 작용으로 이루어지며, 사용자가 임의로 위치한 조작 꼭지점의 움직임을 반영하면서도 표면 형태를 의미하는 운동 특성 벡터를 유지하도록 전체 꼭지점의 위치를 결정한다.

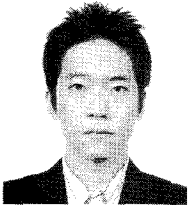
전산 모의 실험을 통해 제안하는 알고리즘이 사용자가 임의로 이동시킨 조작 꼭지점의 위치를 만족시키면서 동시에 메쉬 데이터의 휘어짐이나 평행 이동, 그리고 관절 특징 등을 보존함을 알 수 있었다. 또한, 시퀀스로 표현되는 3차원 영상 정보를 에디팅 기법으로 생성할 수 있는 가능성을 보였다. 따라서 제안하는 알고리즘은 단일 메쉬가 아닌 시퀀스로 이루어진 게임이나 애니메이션을 제작하는데 활용이 가능할 것으로 기대된다.

참고 문헌

- [1] M. Alexa, "Differential coordinate for local mesh morphing and deformation," *The Visual Computer*, vol. 19, no. 2-3, pp.105-114, May 2003.
- [2] H. Fu and C. Tai, "Mesh editing with affine-invariant Laplacian coordinates," *Tech. Report of Hong Kong University of Science and technology*, Aug. 2000.
- [3] T. W. Sederberg and S. R. Parry, "Free-form deformation of solid geometric models," in *Proc. ACM SIGGRAPH*, Aug. 1986, pp. 151-160.
- [4] S. Yoshizawa, A. G. Belyaev, and H. P. Seidel, "Free-form skeleton-driven mesh deformations," in *Proc. ACM Symposium on Solid Modeling and Applications*, June 2003, pp. 247-253.
- [5] K. G. Kobayashi and K. Ootsubo, "t-FFD: Free-form deformation by using triangular mesh," in *Proc. ACM Symposium on Solid Modeling and Applications*, June 2003, pp. 226-234.
- [6] Y. Lipman, O. Sorkine, D. Cohen-Or, D. Levin, C. Rössl, and H.P. Seidel, "Differential coordinates for interactive mesh editing," in *Proc. Shape Modeling International*, June 2004, pp. 181-190.
- [7] A. Sheffer and V. Kraevoy, "Pyramid coordinates for morphing and deformation," in *Proc. International Symposium of 3D Data Processing, Visualization, and Transmission*, Sept. 2004, pp. 68-75.
- [8] R.W. Sumner, M. Zwicker, C. Gotsman, and J. Popović, "Mesh-based inverse kinematics," *ACM Trans. Graph.*, vol.24, no. 3, pp. 488-495, July 2005.
- [9] P. V. Beek, A. Tekapl, and A. Puri, "2-D mesh geometry and motion compression for efficient object-based video representation," in *IEEE International Conference on Image Processing*, pp. 26-29, Oct. 1997.
- [10] J. E. Lengyel, "Compression of time-dependent geometry," in *Proc. ACM Symposium on Interactive 3D graphics*, pp. 89-95, Apr. 1999.
- [11] J.-H. Yang, C.-S. Kim, and S.-U. Lee, "Compression of 3-D triangle mesh sequences based on vertex-wise motion vector prediction," in *IEEE Trans. Circuits Syst. Video Technol.* vol. 12, pp. 1178-1184, Feb. 2002.

저 자 소 개

이 순 영



- 2004년 8월 : 서울대학교 전기공학부 졸업(학사)
- 2007년 2월 : 서울대학교 전기공학부 졸업(석사)
- 2007년 3월 ~ 현재 : 서울대학교 전기공학부 박사과정
- 주관심분야 : 영상 신호 처리, 3D 그래픽스

김 창 수



- 1994년 2월 : 서울대학교 전기공학부 졸업(학사)
- 1996년 2월 : 서울대학교 전기공학부 졸업(석사)
- 2000년 8월 : 서울대학교 전기공학부 졸업(박사)
- 2000년 7월 ~ 2001년 11월 : University of Southern California, Visiting Scholar
- 2001년 12월 ~ 2003년 7월 : 서울대학교 전기공학부 박사후연구원
- 2003년 8월 ~ 2003년 7월 : The Chinese University of Hong Kong, Assistant Professor
- 2005년 8월 ~ 현재 : 고려대학교 전기전자전파공학부, 부교수
- 주관심분야 : 영상처리, 비디오 부호화, 3차원 모델 부호화

이 상 욱



- 1973년 2월 : 서울대학교 졸업(학사)
- 1976년 5월 : Iowa State University 석사
- 1980년 2월 : University of Southern California 박사
- 1983년 3월 ~ 현재 : 서울대학교 전기공학부 정교수
- 2007년 1월 ~ 현재 : IEEE Fellow
- 주관심분야 : 신호 처리, 비디오 부호화, 컴퓨터 비전