

# 특징점 기반의 표면 라이트 필드 몰핑

정은희 안민수 윤민철 이승용

포항공과대학교 컴퓨터공학과

{chilly51, atom, mincheol, leesy}@postech.ac.kr

## Feature-Based Surface Light Field Morphing

Eunhee Jeong Minsu Ahn Mincheol Yoon Seungyong Lee

Dept. of Computer Science & Engineering, POSTECH

### 요약

표면 라이트 필드는 각 시점에서 관찰된 서로 다른 물체 표면의 색 정보를 메쉬에 저장함으로써 물체 표면을 시점 변화에 따라 사실적으로 렌더링할 수 있다. 본 논문에서는 표면 라이트 필드에 대한 영상 기반 편집 기술로서 몰핑 기법을 제시한다. 표면 라이트 필드를 몰핑하기 위해서는 중간 물체의 표면 라이트 필드를 위한 기하 정보와 라이트 필드를 생성해야 한다. 중간 물체의 기하 정보는 메쉬 몰핑을 통해 얻을 수 있다. 중간 물체의 라이트 필드는 두 입력 라이트 필드에서 필요한 정보를 얻어 시점과 기하 정보의 변화에 따라 변형한 후 이를 보간하여 주어진 시점에서의 라이트 필드를 동적으로 얻어낸다. 메쉬 몰핑을 통해 얻어진 중간 물체의 메쉬는 입력 물체에 비해 매우 복잡한 연결 구조를 가지므로 렌더링 속도를 향상시키기 위한 방법을 제시한다. 먼저 메쉬 몰핑 과정에서 메타 메쉬를 만들 때 가까이 있는 정점들을 병합하여 보다 단순한 메타 메쉬를 생성하고 중간 물체를 렌더링하기 위해 메타 메쉬를 사용하지 않고 메타 메쉬를 근사하도록 두 입력 메쉬를 변형한 후 이를 렌더링에 사용한다.

## 제 1 절 서론

일반적으로 컴퓨터 그래픽스에서는 다면체 또는 볼륨과 같은 삼차원 기하모델을 이용하여 렌더링을 한다. 하지만 이와 같은 렌더링 기법은 아무리 정교한 모델을 사용하더라도 실제 물체가 가지는 사실감을 나타내기 어렵다. 실제감이 있는 영상을 빠르게 렌더링하기 위하여 현실에서 획득한 영상 집합으로부터 다양한 시점의 영상을 생성하는 여러가지 영상 기반 모델링 및 렌더링 방법이 연구되어 왔다 [2, 13, 3, 9]. 이와 같은 영상 기반 모델링 및 렌더링 방법은 기존의 비전, 그래픽 기술로는 모델링하기 어려운 복잡한 빛 조건 환경 하의 복잡한 표면 특성을 지닌 물체들에 대해서도 사실적인 렌더링을 할 수 있다. 현재 2차원 영상을 이용하는 방법 뿐만 아니라 물체로부터 기하학적 정보를 추출하여 사용하는 등 다양한 알고리즘이 연구되고 있다 [2, 13].

실제 환경으로부터 얻어진 영상 기반 모델은 일반적으로 변화가 없는 환경이나 물체를 렌더링하는데 사용된다. 그런데 이를 변형함으로써 보다 다양한 렌더링 효과를 얻기 위한 연구도 시도되었다 [13, 15]. Wood 등 [13]은 간단한 메쉬와 물체 표면의 라이트 필드로 이루어진 표면 라이트 필드(surface light field)를 제시하였으며 표면 라이트 필드로 표현된 물체의 변형(deformation)을 보였다. Zhang 등 [15]은 라이트 필드(light field) [9] 및 루미그래프(lumigraph) [3]로 표현된 삼차원 물체의 몰핑(morphing) 기법을 제시하였다. 이와 같은 변형 및 몰핑 방법은 영상 기반으로 표현된 물체를 위한 편집 기술로, 입력 영상 기반 모델로부터 만들 수 없는 새로운 물체의 렌더링을 얻을 수 있었다.

본 논문에서는 이와 같은 영상 기반 편집 기술로 표면 라이트 필드 [2]에 대한 특징점 기반의 몰핑 기법을 제시하고자 한다. Zhang 등 [15]이 라이트 필드에 대한 몰핑 기

법을 제시하였으나 라이트 필드와 표면 라이트 필드는 물체를 표현하는 방식이 다르므로 표면 라이트 필드를 몰핑하기 위해서는 새로운 방법이 필요하다. 표면 라이트 필드는 여러 시점에서 관찰되는 빛 정보만을 저장한 라이트 필드와는 달리 물체 표면의 모든 정점에 대해서 모든 방향에서 관찰되는 빛 정보를 정의하기 위해 빛 정보 뿐만 아니라 물체의 기하 정보인 삼각 메쉬도 함께 저장한다 [13]. 따라서 표면 라이트 필드로 표현된 삼차원 물체의 몰핑을 정의하기 위해서는 기하 정보와 라이트 필드를 동시에 고려해야 한다. 본 논문에서는 몰핑을 통해 얻어지는 중간 표면 라이트 필드를 정의하고 이를 생성하기 위한 방법을 제시한다. 중간 표면 라이트 필드를 위한 메쉬는 기존의 삼각 메쉬 몰핑 방법 [6, 7, 1, 5, 8]을 통해 생성할 수 있으나, 중간 표면 라이트 필드를 위한 라이트 필드를 생성하기 위해서는 새로운 알고리즘이 필요하다. 본 논문에서는 몰핑 생성 알고리즘 뿐만 아니라 실시간으로 중간 물체를 렌더링하기 위하여 렌더링 가속화 알고리즘을 제시한다.

**중간 라이트 필드 근사:** 중간 물체는 메타 메쉬를 이용하여 생성한다. 메타 메쉬를 이용하여 생성한 중간 메쉬는 입력 메쉬와 다른 연결 구조를 가지고 있으며 몰핑 과정 중 표면의 위치가 이동함에 따라 입력 메쉬와는 다른 정점의 위치와 시점 방향을 갖는다. 따라서 입력 표면 라이트 필드에 정점의 연결 구조 변화와 시점 변화를 반영하여 중간 메쉬의 표면 라이트 필드를 재정의하여야 한다. 본 논문에서는 중간 메쉬의 표면 라이트 필드 모두를 생성하는 것이 아니라 입력 표면 라이트 필드로 부터 필요한 정보만을 가져와서 주어진 시점에서의 중간 표면 라이트 필드를 근사하는 알고리즘을 제시한다.

**렌더링 가속화:** 몰핑 과정 중 생성된 중간 표면 라이트 필드는 두 입력 표면 라이트 필드의 속성을 모두 가지

고 있기 때문에 많은 정보를 필요로 한다. 따라서 두 입력 표면 라이트 필드를 렌더링하는 것보다 더 많은 시간이 필요하다. 본 논문에서는 이러한 문제를 해결하기 위해서 단순 메타 메쉬를 만드는 알고리즘과 중간 표면 라이트 필드의 렌더링을 가속화하는 알고리즘을 제시하였다.

본 논문은 다음과 같이 구성된다. 먼저 2 절에서 표면 라이트 필드 물평과 관련된 기존 연구를 살펴 본 후, 3 절에서는 전체 과정에 대하여 설명하겠다. 4 절과 5 절에서는 중간 표면 라이트 필드를 표현하는 중간 메쉬를 생성하는 방법과 주어진 시점에서의 중간 라이트 필드 맵 조각을 생성하는 방법을 차례로 살펴 본다. 6 절에서는 중간 표면 라이트 필드 조각을 렌더링하는 과정을 설명하고 7 절에서는 물평 예제를 살펴 본다. 마지막으로 8 절에서 향후 연구에 대해 논의하고 결론을 맺겠다.

## 제 2 절 기존 연구

Levoy와 Hanrahan [9]은 두 평면 사이의 빈 공간을 지나서 모든 광선에 대한 5차원의 파라미터를 가진 플레네틱 함수(plenoptic function)를 4차원 함수로 표현하여 임의의 시각에서 바라본 광선의 파라미터를 효율적으로 생성하는 방법을 제안하였다. 이 기법은 물체의 기하 정보 없이 샘플링된 광선에서 광휘를 얻어 2차원 영상의 형태로 표현하였다. 새로운 영상은 샘플링된 광선 간의 보간을 통해 계산하였다. Gortler 등 [3]은 두 평면 파라미터화 기법과 유사한 방법인 루미그래프를 제안하고 이를 이용하여 영상을 생성하는 방법을 제안했다.

이전 방법에서는 방법들이 표면의 기하 정보와 독립적으로 라이트 필드 데이터를 표현 한 것과는 달리 [13, 2]는 라이트 필드와 더불어 표면의 기하 정보를 제공하여 물체의 모델링과 에디팅에 효율적인 방법을 제안하였다. 표면 라이트 필드는 모든 시점에 대하여 모든 표면에서 방출되는 광휘를 나타내는 4차원 함수  $f(r, s, \theta, \phi)$ 로 표현되며 이것을 효과적으로 표현하기 위한 기법들을 제시하고자 한다. 여기서 변수쌍  $(r, s)$ 는 표면 상의 위치를 의미하며 다음 변수쌍  $(\theta, \phi)$ 는 시점 방향을 나타낸다.

Wood 등 [13]은 복잡한 기하 정보를 가진 물체의 각 정점에 그 정점에서 관측되는 모든 방향의 광휘 정보를 저장하는 루미스피어(lumisphere)를 정의하는 것을 통해 표면 라이트 필드 함수를 표현 하였다. Wood 등이 제안한 루미스피어 기반의 표면 라이트 필드가 소프트웨어 기반의 실시간 렌더링 알고리즘인 반면 Chen 등 [2]은 하드웨어 기반의 실시간 렌더링이 가능한 표현법인 라이트 필드 맵을 제안 하였다. 이들은 수식 (1)과 같이 적은 수의 기초 영상(basis image) 간의 선형 결합(linear combination)으로 라이트 필드 함수를 근사하는 방법을 제안하였다. 이 방법은 가속화된 그래픽스 파이프라인을 이용하기에 적당한 최적화된 표현법이다. 이는 라이트 필드를 분해(factorization)하여 함수  $g_k(r, s)$ 와  $h_k(\theta, \phi)$ 로 나타내고 텍스처 맵과 같은 샘플링 형태로 저장한다. 이것을 라이트 필드 맵(light field map)이라고 부르고 라이트 필드 맵으로 렌더링하는 과정을 라이트 필드 매핑이라고 한다.

$$f(r, s, \theta, \phi) \approx \sum_{k=1}^K g_k(r, s) h_k(\theta, \phi) \quad (1)$$

여기서 함수  $g_k(r, s)$ 는 표면 맵(surface map)을,  $h_k(\theta, \phi)$ 는 뷰 맵(view map)을 나타낸다.

## 제 3 절 표면 라이트 필드 물평

### 3.1 입력 데이터

본 논문에서는 여러가지 표면 라이트 필드 표현 방법 중에서 [2]가 제안한 라이트 필드 맵으로 표현된 3차원 물체를 입력으로 한다. 이때 3차원 물체의 기하 정보는 모든 면이 삼각형인 삼각 메쉬로 나타내겠다. 삼각 메쉬  $M$ 은  $(P, K)$ 로 표현할 수 있다 [12]. 이때  $P$ 는 메쉬를 구성하는 정점의 좌표를,  $K$ 는 정점들의 연결 관계를 나타내는 단순 복합체(simplicial complex)를 나타낸다.

입력 표면 라이트 필드  $SLF_S$ 와  $SLF_T$ 는 라이트 필드  $L_S, L_T$ 와 메쉬  $M_S, M_T$ 로 이루어져 있다 [2]. 따라서 표면 라이트 필드 물평은 원시 메쉬  $M_S = (P_S, K_S)$ 와 입력 라이트 필드  $L_S$ 로 표현된  $SLF_S$ 와 목표 메쉬  $M_T = (P_T, K_T)$ , 목표 라이트 필드  $L_T$ 로 표현된  $SLF_T$ 가 주어졌을 때, 변환을 0과 1 사이에 있는 중간 메쉬  $M_I = (P_I, K_I)$ 와 중간 라이트 필드  $L_I$ 를 가지는 중간 표면 라이트 필드  $SLF_I$ 를 정의하는 것이다. 변환을 0과 1은 각각  $SLF_S, SLF_T$ 를 나타낸다.

본 논문에서는 중간 표면 라이트 필드를 만들지 않고, 입력 표면 라이트 필드  $SLF_S$ 와  $SLF_T$ 에서 필요한 조각만을 가져와 변형시킨 후 블렌딩하여 주어진 시점에서의 중간 표면 라이트 필드의 영상을 동적으로 얻어 낸다.

중간 메쉬를 만들기 위해서는 대표적으로 리메싱(remeshing) [11]을 하는 방법과 메타 메쉬 [7]를 생성하는 방법 등이 있다. 리메싱으로 중간 메쉬를 만들 경우, 중간 메쉬는 입력 메쉬의 연결 구조를 전혀 포함하고 있지 않기 때문에 입력 라이트 필드 맵에서 중간 물체의 라이트 필드를 얻기가 힘들다. 따라서 본 논문에서는 입력 메쉬의 연결 구조를 모두 포함하고 있는 메타 메쉬를 사용하여 중간 물체를 정의한다.

### 3.2 전체 과정

그림 1과 같이 본 논문에서 제시한 표면 라이트 필드 물평 방법은 크게 세 부분으로 구성되어 있다. 첫번째 부분에서는 중간 메쉬  $M_I$ 를 정의하기 위한 메타 메쉬를 생성한다. 앞서서도 언급했듯이 메쉬는 정점 위치 정보와 연결 관계 정보로 이루어지며, 메타 메쉬  $M_U$ 은 두 메쉬  $M_S$ 와  $M_T$ 의 연결 관계를 결합시킨 것으로 중간 메쉬  $M_I$ 의 연결 관계를 정의한다. 우선 메타 메쉬를 생성하기 위해서는 두 메쉬 간의 대응 관계를 설정해야 한다. 본 논문에서는 두 메쉬 간의 대응 관계를 설정하기 위해 사용자에게 해설된 특징점 간의 대응 관계와 공동 영역으로의 매핑을 사용하였다. 메타 메쉬  $M_U$ 을 통해 중간 메쉬를 정의할 수 있으나 [7]의 Table 1에서 알 수 있듯이 공동 영역으로의 매핑으로 생성한 메타 메쉬는 입력 메쉬보다 다섯 배에서 열 배 정도의 많은 삼각형을 가지게 된다. 실제 라이트 필드 매핑은 텍스처 형태로 인코딩된 표면 맵과 뷰 맵에서 해당 면의 표면 면(표면 맵의 면)과 뷰 면(뷰 맵의 면)을 구한 후 이것을 매핑하는 과정으로 이루어진다. 따라서 렌더링 속도에는 메쉬의 연결 정보 크기에 의존하며 면의 개수가 증가할수록 렌더링 속도도 크게 증가하게 된다. 결국 복잡한 연결 구조를 갖는 메타 메쉬를 렌더링하는 것은 입력 메쉬를 렌더링하는 것보다 더 많은 시간을 요구한다. 본 논문에서는 렌더링 시간을 줄이기 위한 한 방법으로써 단순 메타 메쉬를 만드는 방법을 제시하고자 한다.

두번째 부분에서는 중간 메쉬의 중간 라이트 필드 맵

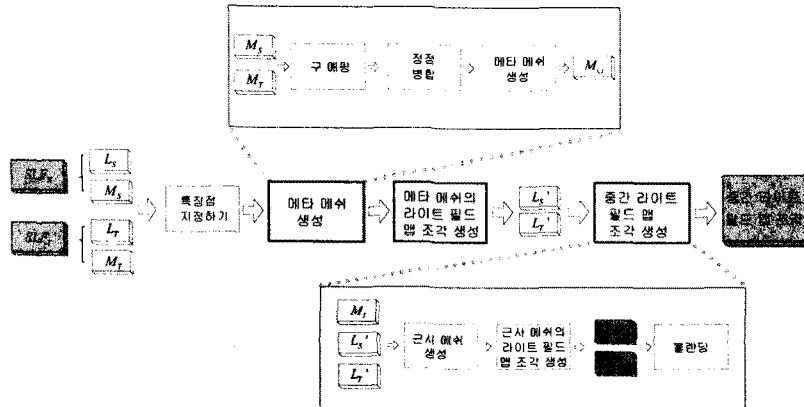


그림 1: 전체 과정

$L_I$ 를 렌더링 하기 위해 입력 라이트 필드 맵에서 필요한 조각을 가져와서 변형시킨 후 블렌딩한다. 표면 라이트 필드는 입력 메쉬의 정점들의 위치와 개수에 따라 달라지므로 중간 물체의 표면 라이트 필드는 입력 메쉬의 기하 정보와 연결 정보가 중간 메쉬로 바뀌는 것을 입력 라이트 필드에 반영하여 얻을 수 있다. 즉 입력 메쉬의 주어진 시점에서의 정점 라이트 필드(vertex light field)에서 얻은 조각을 대응되는 중간 메쉬 면들에 매핑하여 주어진 시점에서의 중간 메쉬의 라이트 필드 맵 조각을 얻는다. 이때 물체 표면 중 정점 위치 및 지역 좌표계의 변화를 동시에 고려해서 입력 라이트 필드 맵 조각을 변형한다.

주어진 시점에서의 중간 물체에 대한 표면 라이트 필드 맵 조각이 얻어지면 마지막으로 중간 표면 라이트 필드 맵 조각을 렌더링한다. 위의 과정을 통해 얻은 변형된 표면 라이트 필드 맵 조각을 그대로 렌더링하면 메타 메쉬를 단순히 만들어도 여전히 복잡하기 때문에 시간이 많이 걸린다. 그래서 본 논문에서는 렌더링 시간을 단축하기 위해서 새로운 알고리즘을 제시한다.

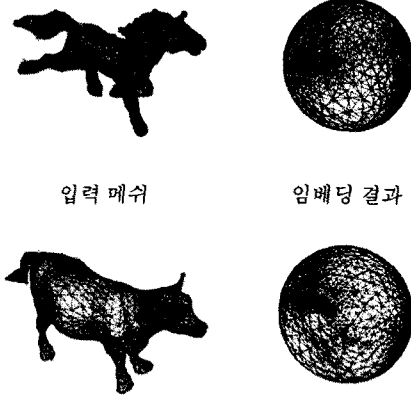


그림 2: 메타 메쉬 생성 과정

## 제 4 절 중간 메쉬 생성

3 절에서 언급했듯이 본 논문에서는 표면 라이트 필드를 표현하고 있는 메쉬 간의 대응 관계를 설정하기 위해서 메타 메쉬를 생성하고 이 대응 관계를 이용하여 메쉬 상의 정점 위치를 보간한다.

### 4.1 메타 메쉬 생성

원하는 대응 관계를 가진 메타 메쉬를 만들기 위해서 먼저 사용자가 두 입력 메쉬  $M_S, M_T$  위의 특징점 간의 대응 관계를 설정한다. 그런 다음, 두 입력 메쉬  $M_S, M_T$  간의 대응 관계를 설정하기 위해서 이들을 공동 영역으로 매핑한다 [7, 1]. 본 논문에서는 공동 영역으로 구를 선택하였고 [1]의 방법을 사용하여 그림 2와 같이  $M_S$ 와  $M_T$ 를 구에 임베딩한 후 특징점이 같은 위치가 되게 변형한다.

$M_S$ 와  $M_T$  위의 정점을 각각  $v_s, v_t$ 라고 하고, 공동 영역 구 상에서 매핑되는 위치가  $p_s, p_t$ 라고 하자.  $v_s$ 와  $v_t$  간의 거리를 구에서  $p_s$ 와  $p_t$  간의 거리로 정의한다.

특징점이 아닌 정점 간의 대응 관계를 설정하는 기본 개념은 공동 영역 상에서 가까운 정점쌍 ( $v_s, v_t$ )를 찾는 것이

다. 그러나 두 메쉬 위의 모든 정점쌍에 대하여 가까운 정점쌍을 찾는 것은 많은 계산량을 요구한다. 이러한 계산량을 피하기 위해서 본 논문에서는 다음과 같은 알고리즘을 제시하였다.

그림 3과 같이  $M_T$  위의 정점  $v_t$ 에 대하여, 공동 영역 구에서  $v_t$ 를 포함하는  $M_S$  위의 면  $f_s$ 을 찾는다. 이렇게 찾아진 면  $f_s$ 의 각 정점  $v_s$ 에 대하여 정점쌍 ( $v_s, v_t$ )을 만들고, 공동 영역 구 상에서 정점쌍 간의 거리가 가까운 것부터 뽑아 일대일 대응 관계를 설정한다. 만일 정점  $v_s$ 가  $M_T$  위의 다른 정점  $v'_t$ 와 대응 관계가 설정되어 있다면,  $v_s$ 와  $v'_t$  사이의 거리가 더 가깝다는 의미이다. 따라서  $v_s$ 와  $v_t$  간의 대응 관계는 설정할 수 없다.

위의 과정을 통해 일대일 대응 관계가 설정된 정점쌍은 공동 영역 구 상에서 같은 위치에 놓여야 한다. 즉  $M_T$  위의 정점  $v_t$ 가 구 상에 임베딩된 위치  $p_t$ 를 수정하여  $v_t$ 와 일대일 대응 관계를 가지는  $v_s$ 의 구 상의 위치  $p_s$ 로 이동한다. 마지막으로 두 메쉬가 구 상에 임베딩된 위치를 이용하여 두 메쉬를 모두 포함하는 메타 메쉬  $M_U$ 을 생성한다. 이러한 과정을 통하여 기존 메타 메쉬 보다 단순한 단순 메타 메쉬를 생성하게 된다.

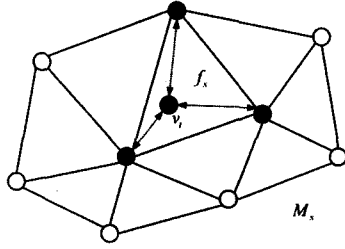


그림 3: 대응관계 설정 과정

## 4.2 중간 메쉬의 정점 위치 결정

4.1 절에서 메타 메쉬  $M_U$ 를 통해 중간 메쉬의 연결 관계를 결정한다. 이제 중간 메쉬를 이루는 정점의 위치를 계산함으로써 중간 메쉬를 결정해야 한다. 본 논문에서는 선형 보간을 통해 정점의 위치를 계산하였다.

공동 영역을 통하여 메타 메쉬는 두 입력 메쉬  $M_S$ 와  $M_T$ 로의 한 지점으로 매핑된다. 즉 메타 메쉬의 모든 정점  $v$ 에 대하여,  $M_S$  상의 위치  $p_s$ 와  $M_T$  상의 위치  $p_t$ 를 가진다. 따라서 주어진 시간  $t$ 에 대하여 정점  $v$ 의 위치  $p$ 는 수식 (2)와 같이 계산할 수 있다.

$$p = (1 - t) \cdot p_s + t \cdot p_t, \quad (0 \leq t \leq 1) \quad (2)$$

## 제 5 절 중간 라이트 필드 맵 조각 생성

본 절에서는 주어진 시점에서의 중간 메쉬  $M_I$ 의 표면 라이트 필드 맵 조각을 생성하는 알고리즘을 설명한다. 중간 라이트 필드를 렌더링 하기 위해서는 중간 메쉬의 모든 정점에 대하여 정점 라이트 필드를 정의해야 한다. 하지만 중간 메쉬  $M_I$ 는 두 입력 메쉬  $M_S, M_T$ 를 모두 포함하는 것으로서 상당히 많은 면과 정점을 가진다. 따라서 정점마다 정점 라이트 필드를 새롭게 정의하는 것은 무척 힘든 작업일 뿐 아니라 메모리 낭비와 렌더링 시간의 증가 등의 어려움을 초래한다. 본 논문에서는 중간 라이트 필드 맵을 새롭게 생성하는 대신 입력 라이트 필드 맵  $L_S$ 와  $L_T$ 에서 주어진 시점에서의 라이트 필드 맵 조각을 가져와 변형한 후 블렌딩하여 중간 라이트 필드 맵 조각을 생성한다.  $L_S$ 와  $L_T$ 를 변형한 라이트 필드 맵 조각을  $L'_S, L'_T$ 이라고 하면, 중간 라이트 필드 맵 조각  $L_I$ 는  $L_I = (1 - t) \cdot L'_S + t \cdot L'_T$ 가 된다.  $L'_S$ 은  $M_S$ 의 기하 정보와 연결 정보가  $M_I$ 로 변형되는 관계 정보를  $L_S$ 에 반영하여 얻을 수 있다.

### 5.1 정점 라이트 필드 매핑

입력 메쉬의 정점  $v_j$ 의 정점 라이트 필드  $f^{v_j}(r, s, \theta, \phi)$ 는  $v_j$ 의 이웃 삼각형(one-ring neighborhood)에 대한 표면 라이트 필드를 무게 중심 좌표로 분할하여 얻어진 표면 라이트 필드이다 [2].  $M_I$ 는  $M_S$ 의 모든 정점을 포함하고 있으므로 이 정점 라이트 필드를 대응되는  $M_I$ 의 표면으로 매핑하여  $L_S$ 를 근사하는  $L'_S$ 을 얻을 수 있다. 즉  $M_S$ 의 한 정점  $v_i$ 와 이웃 정점  $v_{j_0}, \dots, v_{j_n}$ 이  $M_I$ 에서 정점  $v'_i, v'_{j_0}, \dots, v'_{j_n}$ 에 대응된다고 하면 정점  $v'_i$ 와  $v'_{j_0}, \dots, v'_{j_n}$ 들이 이루는 영역을 근사하는 라이트 필드 맵은  $v_j$ 의 정점 라이트 필드로 정의할 수 있다. 이 과정을 정점 라이트 필드 매핑이라고 한다.  $f^{v_j}(r, s, \theta, \phi)$ 는 수식 (1)과 같이 기본 단위, 함수의 곱으로 근사되어 진다. 따라서 정점 라이트 필드 매핑은 표면 맵 매핑과 뷰 맵 매핑으로 나뉘어 진다. 표면 맵  $g'_i$ 은 정

점이나 카메라의 위치와 상관 없이 항상 동일하지만 뷰 맵  $h'_i$ 은 시점 방향에 영향을 받는다. 그림 4와 같이 입력 메쉬와 비교하여 중간 메쉬의 정점의 위치가 변하였으므로 정점의 지역 좌표계와 시점 방향이 모두 변하였다. 따라서 표면 맵 매핑은  $L_S$ 의 표면 면을  $M_I$ 에 대응되는 영역에 동일하게 매핑하지만, 뷰 맵 매핑에서 매핑 되는 뷰 면은 물체 매핑 과정에 따라 변한 시점  $d'$ , 지역 좌표계  $x'$ 와  $y'$ 를 이용하여 구한다. 따라서, 중간 메쉬  $M_I$ 의 뷰 면은 다음과 같은 식으로 계산된 뷰 맵의 텍스처 좌표  $x, y$ 로부터 구한다 [2].

$$x = \frac{(d' \cdot x' + 1)}{2}, \quad y = \frac{(d' \cdot y' + 1)}{2}$$

여기서  $d'$ 은  $v'_i$ 의 시점 방향,  $x'$ 와  $y'$ 는  $v'_i$ 의 변화된 지역 좌표계이다.

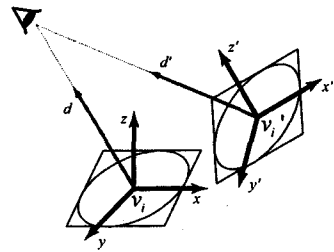


그림 4: 정점 위치 이동으로 인한 라이트 필드 맵의 변화

### 5.2 정점 라이트 필드 분할

$M_S$ 의 한 정점  $v_i$ 의 이웃 삼각형이 이루는 영역은  $M_I$ 에서 여러 개의 면으로 분할되어 진다. 그래서  $M_I$ 의 각 면에 대한 표면 맵과 뷰 면을 얻기 위해서는 정점 라이트 필드 매핑으로 미핑되어진  $L_S$ 의 표면 면과 뷰 면을 분할하여야 한다.  $M_I$ 는  $M_S$ 의 연결 정보를 모두 포함하면서 더 작은 면으로 나뉘어진 메쉬이다. 따라서  $M_S$ 의 면  $f_i$ 가  $M_I$ 에서 여러 개의 면  $f_{ij}$ 로 분할되어 진다면  $f_i$ 의 표면 면은  $f_i$ 에 대한  $f_{ij}$ 의 각 정점의 무게 중심 좌표를 사용하여  $f_{ij}$ 의 표면 면으로 분할할 수 있다. 즉  $f_i$ 의 표면 면에서  $f_{ij}$ 가 차지하는 부분의 텍스처 좌표를 무게 중심 좌표를 이용하여 얻을 수 있다. 표면 면과 달리 뷰 면은 매핑된 뷰 면을 분할하여 얻을 필요가 없다. 뷰 맵은 정점의 이웃 삼각형이 이루는 영역의 모든 지점에 대해 동일한 시점 방향에서 관측되는 광휘를 하나의 광휘로 근사하여 표현하기 때문에  $f_i$ 의 뷰 면을 얻어  $f_{ij}$ 의 뷰 면으로 분할하지 않고 뷰 맵에서 직접  $f_{ij}$ 의 뷰 면을 얻어 온다.

## 제 6 절 렌더링 가속화

본 절에서는 본 절에서는 중간 메쉬  $M_I$ 의 표면 라이트 필드 맵의 렌더링 속도를 향상 시키기 위한 알고리즘을 제시 하도록 하겠다. 3 절에서 언급했듯이 표면 라이트 필드의 렌더링 속도는 메쉬의 연결 정보 크기에 의존하기 때문에 입력 메쉬에 비해 복잡한 연결 구조를 가진 중간 메쉬를 라이트 필드로 렌더링하게 되면 속도가 매우 느려지게 된다. 이를 해결하기 위해 본 논문에서는 복잡한 연결 구조를 갖는 중간 메쉬를 사용하는 것 대신 중간 메쉬의 기하 정보를 근사하면서 입력 메쉬와 같은 연결 관계를 가진 근사 메쉬  $M'_S, M'_T$ 를 사용한다. 근사 메쉬  $M'_S, M'_T$ 는 각각 입력

메쉬  $M_S$ ,  $M_T$ 와 유사한 연결 구조를 갖고 대응되는 중간 메쉬 정점에서 위치 정보를 가져온 메쉬이다. 그리고  $M_S$ ,  $M_T$ 의 라이트 필드 맵은 5 절에서 설명한 방법에 의해 얻는다.

두 근사 메쉬  $M'_S$ ,  $M'_T$ 는 다른 연결 구조를 가지는 입력 메쉬를 이용하여 중간 메쉬를 근사한 메쉬이기 때문에 라이트 필드 보간 과정에서 부자연스러운 윤곽선이 만들어지게 된다. 그 결과 그림 5(a)와 같이 윤곽선 부분이 단순해지고 보간 결과 그림 5(b)와 같이 윤곽선 부분이 서로 일치하지 않게 된다. 따라서 근사 메쉬의 윤곽면을 정제(refinement)하여 두 근사 메쉬의 윤곽선을 일치시켜야 한다. 현재 시점에서 존재하여 윤곽면을 정제하는 다양한 알고리즘들이 소개되고 있다 [14, 4, 10]. 이러한 방법들은 정제 과정에서 메쉬의 연결 구조를 수정해야 하고 두 메쉬의 윤곽선을 일치시키며 정제하기가 어렵다. 본 논문에서는 메타 메쉬를 이용하여 근사 메쉬의 윤곽면의 연결 구조를 수정하지 않고 렌더링 과정에서 근사 메쉬의 윤곽선을 일치시킨다. 즉 렌더링 이전에 근사 메쉬의 윤곽면들을 찾은 후, 렌더링 과정에서 윤곽면인 경우는 윤곽면이 분할되어 얻어진 중간 메쉬  $M_I$  면들의 위치 정보와 라이트 필드 정보로 렌더링한다. 그렇지 않은 경우에는 근사 메쉬  $M'_S$  면의 기하 정보와 라이트 필드 정보로 렌더링한다.

그림 5(c)는 이와 같은 방법으로 얻어진 중간 메쉬의 윤곽면이다. 이때 중간 메쉬 면들로 이루어진 부분과 근사 메쉬를 이루는 윤곽면 사이의 불일치로 인하여 빈 틈이 생기거나 강조되는 부분이 발생한다. 이를 해결하기 위해서 그림 5(d)와 같이 메쉬를 이루는 정점의 위치를 이동시킨다. 즉 메쉬  $M'_S$ ,  $M'_T$ 의 윤곽면의 에지 중 윤곽선이 아닌 에지 위에 있는 중간 메쉬의 정점들을 대응되는 근사 메쉬의 에지 위로 이동시켜줌으로써 이 문제를 해결한다. 이렇게 얻어진 변형 라이트 필드 맵 조각  $L'_S$ 와  $L'_T$ 을 보간하여 중간 라이트 필드 맵 조각  $L_I$ 를 얻는다. 이때  $L'_S$ 와  $L'_T$  각각에 대한 영상을 생성하여 보간하는 것 대신 한 영상에  $L'_S$ 와  $L'_T$ 을 차례대로 렌더링한다. 즉  $L'_S$ 와  $L'_T$ 을 각각 렌더링할 때 주어진 시간  $t$ 에 대한 블렌딩 파라미터를 준다.

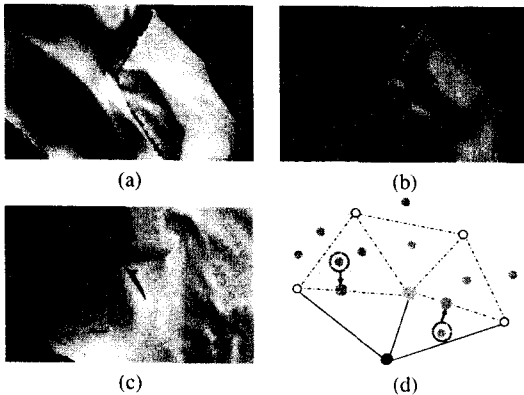


그림 5: (a) 윤곽선 단순화: 붉은 점선은 중간 메쉬의 윤곽선이다. (b) 윤곽선 불일치: 붉은 점선은 윤곽선의 차이를 표시한 것이다. (c) 빈 틈 또는 겹친 영역: 근사 메쉬의 면과 중간 메쉬의 면들의 기하 영역이 정확히 일치하지 않아 생긴다. (d) 중간 메쉬의 정점이 근사 메쉬의 에지 위로 가도록 이동: 푸른 선으로 이루어진 면은 중간 메쉬의 면들이다.

## 제 7 절 실험 결과

본 논문에서 제시한 알고리즘을 이용해서 주어진 세 쌍의 표면 라이트 필드에 대한 자연스러운 물평 애니메이션을 얻었다. 그림 6은 말에서 소로 변하는 결과이고 그림 7은 석상에서 토끼로 변하는 결과이며 그림 8은 룩에서 폰으로 변하는 결과이다. 말 모델은 실제 물체의 표면에서 획득한 영상으로부터 얻어진 라이트 필드 맵으로 렌더링한 것이고, 석상, 토끼, 룩, 폰의 라이트 필드 맵은 인조 물체의 표면에서 얻은 영상으로 만들어진 라이트 필드 맵이다.

표 1은 기존 연구로 구해진 메타 메쉬의 면의 수가 입력 메쉬보다 5 ~ 10배 증가 하는 것에 비해 본 논문에서 제시한 알고리즘으로 생성된 단순 메타 메쉬의 면의 개수는 3 ~ 5배 증가 한 것을 보여 준다. 표 2는 입력 메쉬와 중간 메쉬의 렌더링 시간을 비교한 것이다. 중간 메쉬를 렌더링하는 시간은 입력 메쉬로 렌더링하는 것보다 많은 시간이 걸리는데 이는 두 개의 라이트 필드 맵을 메모리에 적재해야 하는 부하와 근사 메쉬를 얻는 과정의 추가적인 부하 때문이다.

메쉬 종류	$M_S$	$M_T$	기존 $M_U$	단순 $M_U$
말 & 소	7,002	5,150	48,174	22,436
석상 & 토끼	4,976	4,004	37,132	16,394
룩 & 폰	2,256	2,400	18,754	6,516

표 1: 각 메쉬의 면의 개수 (단위: 개)

메쉬 종류	$M_S$	$M_T$	$M_I$
말 & 소	60 ~ 66	60 ~ 66	3 ~ 6
석상 & 토끼	60 ~ 66	60 ~ 66	5 ~ 9
룩 & 폰	60 ~ 66	60 ~ 66	12 ~ 23

표 2: 표면 라이트 필드 렌더링 속도 (단위: FPS), 펜티엄IV 2.0GHz PC에서 실행되었다

## 제 8 절 결론

본 논문은 두 입력 표면 라이트 필드 간의 물평 기법을 제시하였다. 본 기법은 크게 메타 메쉬 생성을 통한 삼차원 위치 보간, 중간 라이트 필드 맵 보간, 그리고 중간 메쉬 라이트 필드 렌더링의 세 부분으로 이루어진다. 중간 메쉬에 대한 중간 라이트 필드는 새롭게 생성되는 것이 아니라 입력 라이트 필드로부터 효과적으로 근사하여 메모리와 계산의 과부하를 해결하였다. 그리고 단순 메타 메쉬를 생성하고 근사 메쉬를 정의하는 것을 통해 렌더링을 가속화하였다.

본 논문에서 제시된 방법은 물평 과정에서 두 물체 간의 광원의 이동을 반영하지 않았다. 표면 라이트 필드는 고정된 환경에서 획득되어지므로 두 표면 라이트 필드의 광원이 서로 다른 위치에 있을 경우 광원이 이동되도록 중간 표면 라이트 필드를 생성하기는 어렵다.

## 감사의 글

본 논문은 BK21 사업을 통하여 포항공과대학교 전자.컴퓨터공학부에 주어진 교육부의 재정 지원과 게임 애니메이션 연구 센터의 재정 지원을 받았습니다.

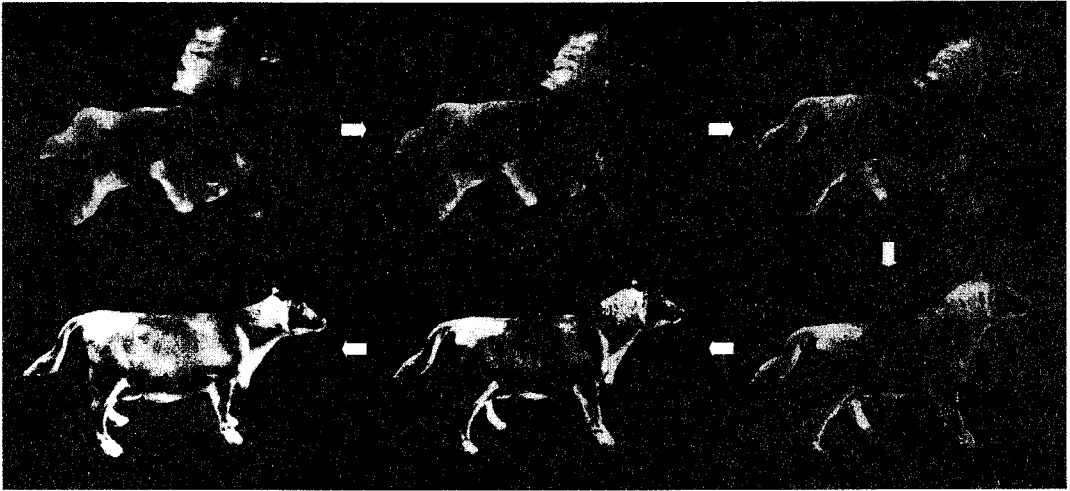


그림 6: 말에서 소로의 변화

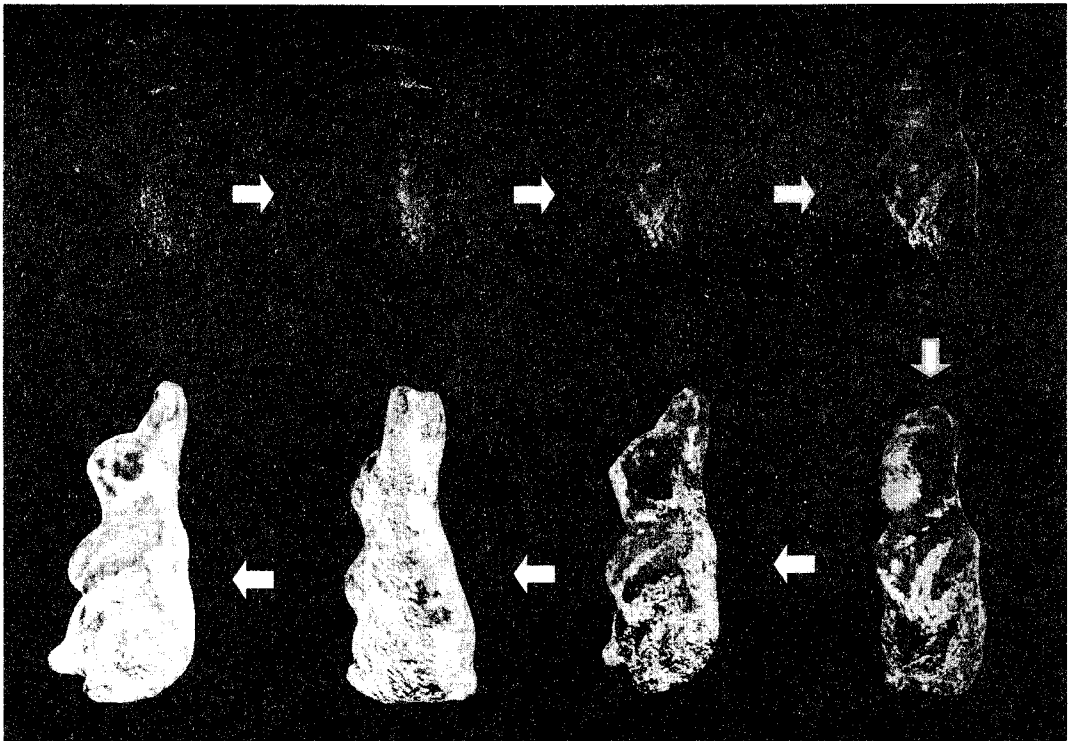


그림 7: 석상에서 토끼로의 변화

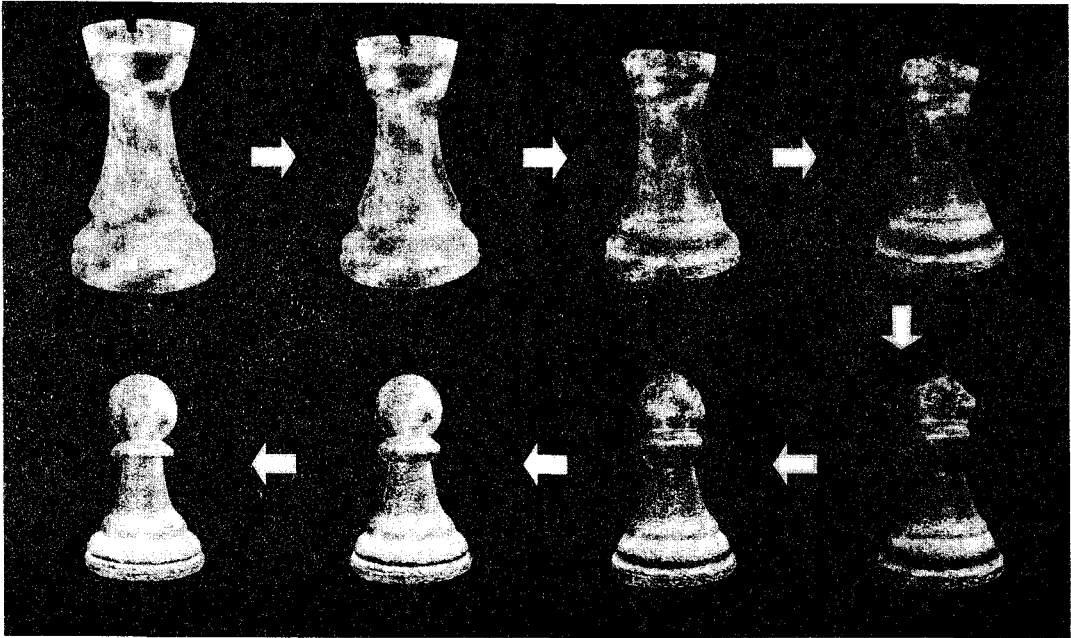


그림 8: 룯에서 폰으로의 변화

## 참고 문헌

- [1] ALEXA, M. Merging polyhedral shapes with scattered features. *The Visual Computer* 16, 1 (2000), 26–37.
- [2] CHEN, W.-C., BOUGUET, J.-Y., CHU, M. H., AND GRZESZCZUK, R. Light field mapping: Efficient representation and hardware rendering of surface light fields. *ACM Computer Graphics (Proc. of SIGGRAPH 2002)* (2002).
- [3] GORTLER, S. J., GRZESZCZUK, R., SZELISKI, R., AND COHEN, M. F. Lumigraph. *Computer Graphics (Proc. SIGGRAPH '96)* (1996), 43–54.
- [4] HOPPE, H. View-dependent refinement of progressive meshes. *ACM Computer Graphics (Proc. of SIGGRAPH '97)* (1997), 189–198.
- [5] HUGHES, J. F. Scheduled fourier volume morphing. *ACM Computer Graphics (Proc. of SIGGRAPH '92)* (1992), 43–46.
- [6] LAZARUS, F., AND VERRAUST, A. Three dimensional metamorphosis: A survey. *The Visual Computer* 14, 8/9 (1998), 373–389.
- [7] LEE, A. W., DOBKIN, D., SWELDENS, W., AND SCHRÖDER, P. Multiresolution mesh morphing. *ACM Computer Graphics (Proc. of SIGGRAPH '99)* (1999).
- [8] LERIOS, A., GARFINKLE, C. D., AND LEVOY, M. Feature-based volume metamorphosis. *ACM Computer Graphics (Proc. of SIGGRAPH '95)* (1995), 449–456.
- [9] LEVOY, M., AND HANRAHAN, P. Light field rendering. *Computer Graphics (Proc. SIGGRAPH '96)* (1996), 31–42.
- [10] LUEBKE, D., AND ERIKSON, C. View-dependent simplification of arbitrary polygonal environment. *ACM Computer Graphics (Proc. of SIGGRAPH '97)* (1997), 199–208.
- [11] PRAUN, E., SWELDENS, W., AND SCHRÖDER, P. Consistent mesh parameterizations. *ACM Computer Graphics (Proc. of SIGGRAPH 2001)* (2001), 179–184.
- [12] SPANIER, E. H. *Algebraic Topology*. McGraw-Hill, 1966.
- [13] WOOD, D. N., AZUMA, D. I., ALDINGER, K., CURLESS, B., DUCHAMP, T., SALESIN, D. H., AND STUETZLE, W. Surface light fields for 3d photography. *ACM Computer Graphics (Proc. of SIGGRAPH 2000)* (2000).
- [14] XIA, J. C., AND VARSHNEY, A. Dynamic view-dependent simplification for polygonal models. *IEEE Visualization '96* (1996), 327–334.
- [15] ZHANG, Z., WANG, L., GUO, B., AND SHUM, H.-Y. Feature-based light field morphing. *ACM Computer Graphics (Proc. of SIGGRAPH 2002)* (2002).