

증강현실에서의 반사구를 활용한 사실적 조명 생성

이석준 정순기

경북대학교 IT대학 컴퓨터학부

sukjuni@vr.knu.ac.kr, skjung@knu.ac.kr

Regeneration of Plausible Lighting using a Specular Sphere in Augmented Reality

Seokjun Lee

Soon Ki Jung

School of Computer Science and Engineering, College of IT Engineering, Kyungpook National University

요약

본 논문에서는 마커리스 증강현실 가시화에서 반사구(mirror-sphere)를 이용하여 실제 환경의 광원위치를 추정된 결과를 증강되는 가상객체에 적용하는 방법을 제안한다. 고정된 환경조명의 안정적인 추정을 위해, 카메라를 반사구 주위를 돌며 촬영하여 반사구에 대한 다중 영상을 획득한 후, 각 광원 벡터 클러스터의 주 광원 방향을 추정한다. 본 논문에서는 환경조명의 광원방향 추정과 함께 움직이는 광원에 대한 광원방향도 추정함으로써, 실제 조명환경을 증강되는 가상객체에 반영하여 사실적인 음영효과와 그림자를 생성한다. 본 논문에서 제안하는 조명 추정 방법은 조명 환경에 대한 어떠한 사전 정보 없이도 즉각적으로 증강현실 가시화에 적용할 수 있다.

Abstract

This paper presents a practical method to estimate the directions of light sources in real environment, using a mirror sphere placed on a set of known natural features in augmented reality. For the stable result of static lighting, we take the multiple images around the sphere and estimate the principal light directions of the vector clusters for each light source in realtime. We also estimate the moving illuminant for changes of the scene illumination, and augment the virtual objects onto the real image with the proper highlighting and shadows. The proposed method of this paper can be applied to augmented reality visualization without any previous information respecting the environmental illuminations.

키워드: 증강현실, 다중광원, 가상조명

Keywords: Augmented Reality, Multiple Illuminants, Virtual Lighting

1. 서론

증강현실(AR, augmented reality)은 가상의 객체를 실사 위에 정합함으로써 실제 환경에 대한 추가적인 정보나 가상의 객체를 효과적으로 가시화할 수 있는 기법이다. 증강현실 분야에서는 현장에 대한 영상에 겹쳐지는(overlay) 가상객체를 사람이 바라보았을 때, 마치 그곳에 있는 것처럼 느낄 수 있도록 하기 위한 다양한 기술적 방법들이 연구되어왔다. 사람이 가상의 객체와 실제 환경을 구별할 때 필요한 다양한 인식적 요소가 사용되지만, 증강되는 객체의 실제감을 느끼는 데 가장 필수적 요소로써, 실제 환경에 대한 카메라의 위치와 자세를

보정하여 가상객체를 배치하는 카메라 정합 기술과 실제 주변 환경과 조화롭게 가상객체를 장면에 그리는 렌더링 기술이 주목 받고 있다.

그래픽스 렌더링에서는 광선추적을 통하여 가상객체를 현실감있게 그리는 과정이 보편화 되었으나 실제 광원을 반영하기 위해 여전히 사람의 입력을 이용하거나 사전에 환경조명에 대한 정밀한 보정을 거쳐야 한다. 장면에 등장하는 조명환경을 획득하여 분석하고 그 결과를 가상객체 렌더링에 반영하는 과정은 사용자로 하여금 컴퓨터가 생성한 가상객체에 대해 실제감을 느낄 수 있게 하는데 있어 매우 중요한 요소이다.

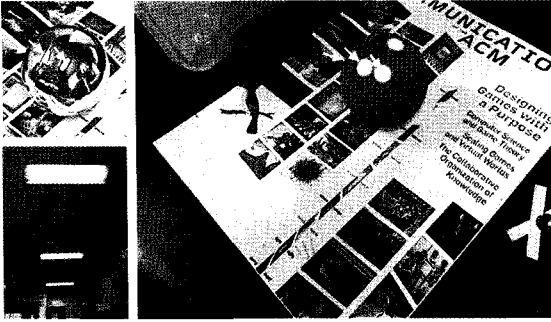


그림 1: 실제 조명환경(좌측 열)의 광원 추정 결과를 증강현실 가상객체에 반영한 예제

본 논문에서는 마커리스(markerless) 증강현실 환경에서 반사구를 이용한 사실적 광원추정기법을 제안한다. 제안하는 방법은 실제 조명환경에 대한 어떠한 사전정보 없이도 수행가능하며, 증강현실 가시화에 즉각적으로 적용할 수 있는 방법이다. 사용자는 증강현실 가시화가 수행되는 구동시간에 간단한 조작을 통해 시스템이 광원의 방향을 추정할 수 있도록 하고, 시스템은 반사구에 상대적인 카메라 위치와 자세에 대한 기하정보를 이용하여 광원의 방향을 추정한 후 가상객체의 렌더링에 반영한다.

본 연구에서 수행한 환경조명 추정과 그 결과를 증강현실에 반영한 예제는 그림 1과 같다. 그림 1은 광원 추정결과가 추정에 사용된 실제 반사구에 작용하는 광원과 같은 조건으로 반영됨을 나타낸 결과이며(노란색 화살표로 표시), 좌측 상단에는 카메라의 실제장면에 나타난 반사구의 스크린 샷이고, 좌측 하단에는 실험이 수행된 실내 공간의 천장 조명을 촬영한 장면이다. 그림 1의 결과에서 나타난 세 개의 정반사 지점(specular spot)은 천장등에 대한 광원 추정결과이며, 나머지 하나는 카메라 시야 반대편에 있는 PC모니터 화면이 반사구에 반영된 결과이다. 맨 바깥쪽 광원은 현재 장면영상에 포착되지 않았으나, 본 논문에서 제안한 순차적 카메라 이동을 통한 광원 획득 기법을 통해 얻을 수 있으며, 그 결과는 본 논문의 7장에서 자세히 나타내었다. 그림에서 나타난 구는 광원 추정을 위해 사용된 반사구와 같은 크기로서, 반사구와 같은 3차원 위치에 배치되었다.

본 논문의 구성은 다음과 같다: 2절에서는 광원을 인식하고 장면의 조명효과를 추정하는 여러 관련연구들을 소개한다. 3절에서는 본 논문에서 제안한 시스템의 전체적인 흐름을 기술하고, 4절에서는 카메라 보정 과정과 장면영상 분석기법을 설명한다. 5절에서는 본 논문에서 제안하는 증강현실 시스템에서의 실제 조명환경 반영을 위한 광원추정기법에 대해 자세히 설명하고, 6절에서는 본 논문의 실험이 수행된 렌더링 환경을 설명한다. 7절에서 실험 내용과 결과를 다루고, 8절에서는 본 연구의 결론과 향후에 필요한 연구에 대해 기술한다.

2. 관련연구

증강현실 기술에서는 가상객체의 사실적인 표현을 위해 다양한 접근법을 통한 노력들이 이루어 지고 있다. 수많은 연구자들이 가상의 객체를 현실감있게 실제 장면에 정합하고, 렌더링하기 위한 기법들이 제안되었는데, 특히 사실적인 렌더링을 위해 실제 장면의 조명환경을 분석하여 가상의 객체에 반영하는 기술은 가장 각광받는 증강현실 가시화 이슈 중 하나이다. 하지만 실시간으로 동작하는 증강현실 환경에서 실제 환경의 조명상태를 계산하는 일은 매우 어렵기도 하다.

카메라의 장면으로부터 실제 환경의 조명상태를 계산하기 위한 노력은 다양한 연구결과로 제시되었는데 [1], 주로 조명 추정을 위한 방법적인 분류로서, 측정용 매개체를 사용한 방법 [2, 3, 4, 5, 6, 7, 8, 9, 10]과 매개체없이 장면만으로 조명환경을 복원하는 방법 [11, 12, 13]으로 나눌 수 있다.

측정용 매개체를 이용하여 조명환경을 획득하는 방법은, 난반사 구형 모델(diffused spherical model)을 사용한 방법 [2, 3, 4, 5]과 정반사 구형 모델(specular spherical model)을 사용한 방법 [6, 7, 8, 9]으로 나눌 수 있는데, 이들의 연구에서 구형 모델을 사용하는 이유는, 대상 객체가 위치하는 지점에 직접적으로 관여되는 환경조명에 대한 상태를 3차원 공간에서 획득하고 분석하기에 가장 유리한 접근법이기 때문이다.

Zhang [2, 3]은 난반사구의 명암변화도를 측정함으로써 실제 환경조명의 조도와 방향을 추정하는 근거로 사용하였으며, Feng [4]는 증강현실 환경에서 임의의 조명이 주어졌을 경우에 위치에 따른 조도의 자연스러운 적용을 위해 두 개의 난반사구를 이용하여 환경조명의 변화를 측정하였다. 이들의 연구에서는 정반사광이 배제된 물체(lambert surface)에 대한 조명도를 측정하고 측정된 명암도의 변화를 가상객체에 적용하여 렌더링 함으로써 보다 자연스러운 명암도 표현이 가능케 하였다. Aittala [5]는 증강현실에서 사실적 렌더링을 구현하기 위해 난반사구를 이용하여 조명의 색상과 명암도를 측정하여 가상의 객체에 적용하는 방법을 제안하였다. 하지만 이들의 방법에서는 정반사 광에 대한 고려없이, 난반사구에 적용되는 주변광의 명암도만을 측정하였고, 광원의 방향이나 관측하는 카메라의 움직임에 대한 고려가 없었다.

정반사 구형 모델을 이용하여 정반사 광을 인식함으로써 광원 상태를 측정하는 연구로는 Debevec [6], Okatani [7], 그리고 Supan [9]의 연구를 들 수 있다. 이들은 고정된 카메라와 사용자 입력으로 주어진 정반사구에 대한 정보를 바탕으로 카메라가 비추고 있는 대상에 환경조명이 미치는 조명상태를 계산하는 연구를 수행하였다. 이 연구에서는 미리 반사구의 위치 정보와 함께 카메라를 정확한 위치에 고정하여 장면을 보정한 후에 보정된 조명정보와 광선추적결과를 장면 렌더링에 활용하였다. 하지만 이들의 방법은 고정된 카메라와 정밀한 실험 환경 설정이 필요하므로, 주로 움직이는 카메라를 사용하는 증강현실 시스템에서는 활용하기 힘들뿐만 아니라, 매우 많은

광학적 추적방식을 사용하므로 계산방법이 복잡하고 그 양이 많아 증강현실 구동시간 내에 사용하기는 어렵다는 단점이 있다. Kanbara [8]의 연구에서는 증강현실에서 흔히 사용하는 사각형 이진마커를 사용하여 카메라의 자세를 추적하고 매우 작은 크기의 정반사 구를 이용하여 광원 샘플을 획득하여 가상 객체 적용하는 방법을 제시하였으나, 반사 구의 크기가 작아 정확한 방향의 광원을 추정하기 보다는 대체적인 광원 효과를 내는데 그쳤고, 직접적인 광원의 방향을 추정하지 않았으므로 정반사 광을 표현하거나 광원 방향에 대한 광원 효과를 내지는 못하였다. 반면에 이들의 연구에서는 반사구를 활용하여 조명의 방향을 추정하고, 추정정보를 활용한 조명환경 복원이 매우 의미 있는 과정임을 증명하였다.

Knecht [10]의 연구에서는 마커가 부착된 손전등을 이용하여 증강현실 가상객체와 상호작용하는 조명효과를 제안하였다. 이 연구에서는 장면에 등장하는 조명의 위치와 자세를 정의하는 마커를 카메라로 인식하여 광원의 위치와 방향을 설정하고, 실제 광원의 세기와 광원의 크기를 어안렌즈를 장착한 별도의 카메라를 이용하여 보정하는 사전과정을 거친다. 설정된 광원정보는 가상의 객체의 조명효과에 사용되며, 실제 조명의 반사광 효과를 증강현실환경에서 구현하였다. 하지만 위 연구에서는 광원을 인식하기 위한 마커가 카메라 장면에 항상 등장하기때문에, 마커에 의한 실제 장면의 가려짐이나 실제 장면과 관련성이 없는 객체의 등장으로 인한 가시적 불편함이 존재한다.

Haller [11]와 Madsen [12]의 연구에서는 광원인식을 위해 특정한 매개체를 사용하는 대신에, 영상에 등장하는 객체의 그림자를 인식한 후 그림자의 길이와 방향정보를 바탕으로 광원을 생성하여 가상의 객체에 적용하는 방법을 제시하였다. 이들의 방법에서는 장면상에 직접적으로 조명을 측정하기 위한 매개체를 등장시키지는 않았지만, 장면상에 강한 그림자를 생성해 낼 수 있는 객체가 존재하여야 하고, 실제 환경에도 강한 그림자를 생성할 수 있는 세기를 가진 광원이 존재하여야 한다는 제약이 존재한다.

Frahm [13]의 연구에서는 마커리스 증강현실기반의 방송에서 실제 환경의 조명효과를 가상의 객체에 적용하기 위해, 촬영장 스튜디오 내부의 광원 방향으로 설치된 별도의 카메라를 이용하여 광원의 위치를 측정하고, 스튜디오 내부를 사전에 3차원으로 복원하여 측정된 광원을 배치함으로써, 증강현실 장면에 등장하는 가상의 객체에 조명효과를 부여하는 기법을 제시하였다. 제시된 방법은 비교적 정확하게 광원위치를 복원하고, 가상객체에 부여된 조명 렌더링의 질이 우수하다는 장점이 있지만, 방송 스튜디오와 같이 고정된 환경에서 사진에 복원된 3차원 공간의 기하정보를 활용할 수 있는 환경에서만 사용이 가능하고, 전체적인 처리과정이 다중의 카메라와 입력정보를 복합적으로 처리하기 때문에 사전처리 없이 실시간으로 사용하기에는 계산량이나 계산시간에서 부담이 많다.

이처럼 다양한 조명환경 분석기법이 제안되고 있음에도 불

구하고 많은 증강현실 응용 프로그램들은 여전히 실제 환경조명에 대한 반영없이 초기에 부여된 인공적인 조명을 사용하거나, 조명을 반영하지 않는 가시화 기법으로 가상의 객체를 렌더링 하여 실제 장면과의 부조화를 이루기도 한다. 증강현실 가시화에 사실적 조명을 부여하기 위해서는 광원에 대한 물리적인 정확도나 영상 화질의 우수성 등의 다양한 요소들이 적절히 조화되어 가상의 객체에 적용되어야 하겠지만, 실재감을 부여하기 위한 물리적 요소들을 모두 충족하기에는 증강현실 응용프로그램이 구동되는 시스템에 따라 사용상에 부담이 많이 따른다. 본 논문에서는 증강현실 가시화에서 사용할 수 있는 사실적 조명을 생성하기 위해, 프로그램 구동시간에 간단하게 수행할 수 있는 계산량으로도 실제 조명환경의 위상적 정확도를 적절히 조화롭게 반영할 수 있는 방법을 제안한다.

3. 시스템 구성

본 논문에서는 실제 환경의 조명상태를 증강현실 가시화에 적용하는 방법을 제안한다. 본 논문에서 제안하는 시스템은 실제 환경에 존재하는 고정된 조명환경을 선 보정하고, 이를 바탕으로 움직이는 조명이 있을 경우에도 동적조명방향을 추정하는 과정을 수행한다. 광원의 방향은 특징점 기반의 카메라 보정으로 추정된 카메라 자세와 반사구 상에 반사된 광원의 하이라이트 점(highlight point)과의 기하학적 관계를 이용하여 계산된다. 광원방향 추정의 에러를 줄이기 위해서는, 반사구 상의 각 하이라이트 점을 이루고 있는 광원방향벡터들을 클러스터링하고, 이를 광원벡터에 대한 후보 벡터로 지정하여 이들의 최적 벡터(optimized vector)을 구함으로써 광원방향 벡터의 에러를 감소시킨다. 광원의 추정결과는 환경조명의 정반사광으로 설정하여 증강현실 가시화에서 가상객체의 명암도(shading)와 그림자 생성에 사용된다. 본 논문에서 제안하는 방법은 그림 2와 같은 순서로 사실적인 조명이 반영된 증강현실 장면을 생성한다.

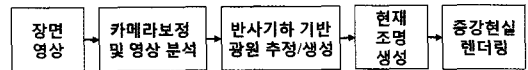


그림 2: 사실적 광원 생성 처리 순서

사용자는 초기화 단계에서 인식 가능한 객체(특징점 마커) 위에 반사구를 올려두고, 반사구를 기점으로 하여 카메라를 이동시키며 장면상에 반사구와 특징점 마커가 꾸준히 등장하도록 촬영을 한다. 시스템은 장면영상이 촬영된 시점의 카메라의 자세를 보정하는 동시에 획득한 장면영상을 분석하여 광원에 대한 후보벡터들을 획득한다. 카메라의 움직임에 따라 획득한 시점이 다른 장면영상으로부터 광원에 대한 후보 벡터 군(vector cluster)을 설정하고 이들 각각에서 최적의 대표 벡터를 계산하여 광원에 대한 방향벡터로 설정한다.

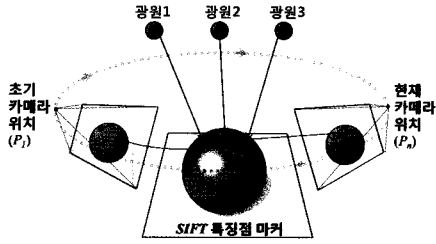


그림 3: 특징점 마커, 반사 구, 광원, 카메라 시점의 변화에 따른 정반사 광의 위상에 대한 기하관계

제안하는 시스템에서는 가상객체에 적용될 실제 광원의 물리적 정확도를 향상시키기 위해 카메라를 이동시키며 다중 시각(multiple view)에 대한 장면영상을 획득하여 이를 광원추정 과정에 이용한다. 이는 그림 3에서 도시된 것처럼, 특징점 마커, 반사구, 반사구 상의 정반사 광, 그리고 카메라 시점간의 기하적 위치관계를 최대한 이용함으로써 가상객체가 렌더링될 위치와 같은 위치에 배치된 반사구에 적용되는 실제 광원을 최대한 반영하기 위함이다 [14]. 최종적으로 추정된 광원 정보는 증강현실에서 가시화될 가상객체에 적용하여 실제 조명환경이 반영된 음영효과와 그림자를 생성한다.

4. 카메라 보정 및 영상 분석

증강현실 시스템에서 일반적으로 가장 먼저 수행하는 과정은 카메라의 자세를 보정하는 과정이다. 카메라 자세보정은 미리 정의된 객체에 대한 모델 좌표계(model coordinate)에서의 카메라 위치(position)와 방향(orientation)을 계산하는 과정을 일컫는다. 본 절에서는 안정적인 카메라 보정을 위한 전반적인 단계를 설명한다.

4.1 카메라 보정을 위한 마커 인식

본 논문에서는 증강현실에서 흔히 사용하는 이진화된 마커가 아닌, 사진인체물과 같이 자연 특징(natural feature)으로 정의된 특징점 마커를 이용하여 카메라 자세를 보정한다. 카메라의 현재 시점에서 나타나는 특징점 마커를 인식하기 위해서, 인식에 사용할 도서의 표지를 SIFT(Scale-Invariant Feature Transform) [15]를 통해 특징점을 추출하여 특징점 마커를 정의하고, 현재 시점에 대한 장면영상과 매칭(matching)을 통해 유사도를 분석하여 장면영상 내에 나타난 객체를 인식한다. 본 논문에서는 GLSL(OpenGL Shading Language)로 구현된 GPU용 SIFT를 이용해 특징점을 분석함으로써, 비교적 느린 SIFT 특징점 분석과 매칭 속도를 향상시키고, 이를 증강현실 시스템에 사용하기에 무난한 속도를 구현하였다. 본 논문에서는 카메라 보정을 위해 그림 1 우측에서 나타난 특정 도서의 표지를 사용하였지만, 실제 시스템 구동에서는 주변 사물에서

흔히 구할 수 있는 평면 객체를 인식시킴으로써 본 논문에서 사용된 도서 표지와 동일한 형태의 역할을 할 수 있다.

4.2 안정적인 광원 획득을 위한 카메라 보정

객체의 인식은 학습된 객체의 특징점과 현재 영상에 나타난 특징점과의 매칭유사도로 분석되는데, 학습객체 특징점의 매칭점이 입력영상에 수렴하는 영역과의 호모그래피(homography)를 이용하여 평면객체의 대응영역을 계산한다. 특징점 분석과 매칭이 완료된 후에는, 사전에 정의된 특징점 마커영역과 현재 장면영상에서의 특징점 마커 영역에 해당하는 특징점의 대응관계로부터 특징점간의 호모그래피를 계산한다. 본 논문에서 안정적인 카메라 보정을 위한 처리 단계는 그림 4와 같이 나타낼 수 있다.

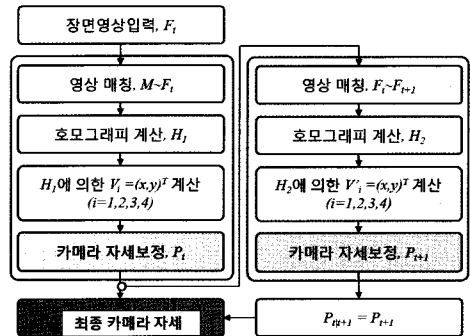


그림 4: 카메라 안정화를 위한 보정 단계

카메라 모션의 안정화를 위해서는 각 영상 프레임으로부터 구한 카메라 자세를 평균화 하는 것으로는 전체적인 카메라 모션의 부드러움을 구현하기에 무리가 있다. 기존의 SLAM(Simultaneous Localization and Mapping)이나 SFM(Structure From Motion)을 이용한 증강현실 시스템과 같이 장면을 미리 3차원으로 복원하는 과정을 거친다면 더할나위없이 정확한 카메라 정합을 구현할 수 있겠지만, 이는 사전에 장면에 대한 정보를 학습하고, 계산 정보를 최적화 하는데 비교적 많은 시간과 비용이 소모된다.

본 논문에서는 이러한 사전정보 없이 부드러운 카메라 모션을 구현하기 위해 영상 프레임간의 호모그래피를 이용하여 카메라의 자세를 안정화한다. 호모그래피를 이용한 영상간의 변환관계는 수식 (1, 2)로 정의되며, 그림 4에서 나타낸 것과 같이 현재 영상의 특징점들을 이용하여 다음영상에서의 위치를 추정하고, 이를 2D-3D점의 DLT(direct linear transformation) [16]를 통해 다음 카메라의 자세를 보정하는 과정을 거친다.

본 논문에서는 기본적으로 기존의 증강현실 시스템에서 보편적으로 사용하는 카메라 자세 추정 기법을 사용한다. 다만, 안정적이고 일관적인 증강현실 내비게이션을 위해 마커와 영상(M-F_t)간의 호모그래피와 영상과 영상(F_t-F_{t+1})의 호모그

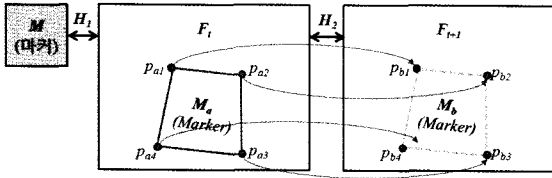


그림 5: 카메라 자세 안정화를 위한 영상 특징 관계정의

래피를 검출함으로써 카메라 자세(P_t)를 안정화 시키고, 때로 현재 카메라 시야에서 기 정의된 마커가 벗어났을 경우에도 위 관계를 통해 카메라 자세($P_t + 1$)를 보상할 수 있다 [17]. 본 논문에서 제안하는 카메라 자세 안정화를 위한 영상 특징관계 정의는 그림 5와 같이 나타낼 수 있다.

대응점 (p_a, p_b)가 주어졌을 때, 호모그래피 (H_{ab})는 다음 식 (1)과 같이 정의된다.

$$p_a = \begin{bmatrix} x_a \\ y_a \\ 1 \end{bmatrix}, p'_b = \begin{bmatrix} wx_b \\ wy_b \\ w \end{bmatrix}, H_{ab} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}, \quad (1)$$

이때, $p'_b = H_{ab}p_a$ 를 만족하고, $p_b = p'_b/w$ 이다. 올바른 호모그래피 계산을 위해 RANSAC(RANdom SAMple Consensus)을 수행하여 최종적인 호모그래피(H)를 계산할 수 있으며, H 를 알면, 아래 수식 (2)에 의해 영상간의 변환관계를 알 수 있다. 영상 간 상동관계를 통한 객체 인식은 H_1, H_2 를 구할 때 각각 동일한 방법으로 적용될 수 있다.

$$\begin{aligned} Z_i &= 1/(x_i h_{31} + y_i h_{32} + h_{33}), \\ X_i &= (x_i h_{11} + y_i h_{12} + h_{13})Z_i, \\ Y_i &= (x_i h_{21} + y_i h_{22} + h_{23})Z_i, \end{aligned} \quad (2)$$

인식하기 위한 객체가 $Z=0$ 인 평면상에 존재하고, 이때 카메라의 CoP(Center of Projection)가 장면평면 상에 존재하지 않는다는 가정 하에 카메라 투영행렬(P , projection matrix)은 식 (3)을 만족한다.

$$p'_b = [P_1 P_2 P_3 P_4] \begin{bmatrix} x_a \\ y_a \\ 0 \\ 1 \end{bmatrix} = [P_1 P_2 P_4] \begin{bmatrix} x_a \\ y_a \\ 1 \end{bmatrix}, \quad (3)$$

카메라의 투영행렬 P 가, 카메라 내부 파라미터(intrinsic parameter) K , 회전행렬(rotation matrix) R , 이동벡터(translation vector) t 로부터 구해질 때, $P = K[R|t]$ 가 성립된다. 카메라 내부 파라미터 K 를 안다고 가정했을 때, H_{ab} 로부터 R_1, R_2, t 를 구할 수 있으며, 식 (1)과 식 (3)에 의해, $H_{ab} = K[R_1, R_2, t]$ 의 꼴로 나타낼 수 있다(R_i 는 R 행렬의 i 번째 열 성분). 다시 말해, 평면 π 상의 점 $x_\pi = (x, y, 1)^T$ 과 투영된 영상의 점 x 사이의 매핑관계는 일반적인 평면 호모그래피인 $x = Hx_\pi$ 로

표현될 수 있으며, 이때 H 는 rank3인 3×3 행렬이다. 따라서 $P = K[R|t]$ 인 카메라 투영행렬이 $Z=0$ 인 평면상의 점으로부터 결정될 때, 앞서 기술한 바와 같이, 사영기하학(projective geometry)의 원리를 이용하면, 장면 영상에서의 2차원 점들에 대응하는 특징점 마커의 2차원 점들간의 호모그래피 H 를 구하고, 호모그래피 H 를 카메라의 투영행렬 P 로 해석할 수 있다. 그 결과로 실세계 좌표계상의 반사 구의 위치에 대응하는 카메라의 위치(t)와 자세(R)를 알 수 있으며, 이를 통해 카메라의 영상평면에 사상되는 반사 구의 영역도 구할 수 있다.

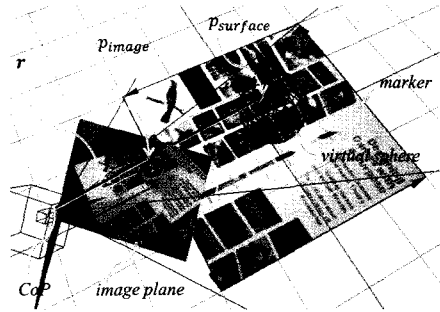


그림 6: 보정된 카메라 자세와 좌표계상의 반사 구 모델

그림 6에서는 카메라 보정과정을 거친 후의 카메라(CoP), 카메라의 영상평면(p_{image}), 특징점 마커, 특징점 마커 위에 올려 둔 반사 구 모델을 OpenGL을 이용하여 3차원 좌표계 상에 표시하였다. 그림 6에서 표시된 것처럼, 카메라(CoP)로부터 영상평면(p_{image}) 상의 하이라이트 영역을 통과하여 가상 구의 표면($p_{surface}$)에 이르는 벡터(r)를 알고, 이 벡터(r)가 구 표면($p_{surface}$)에 의해 반사되는 벡터를 계산함으로써 광원의 위치를 추정할 수 있게 된다. 반사 구는 특징점 마커상에 고정되어 있으므로, 특징점 마커와 카메라간의 좌표계상의 기하정보만 알면 그래픽스에서 흔히 사용하는 반사기하를 이용하여 반사 광을 추정할 수 있게 된다.

4.3 반사 광 영역 추출

카메라 보정이 완료되면 그림 7과 같은 순서로 장면 영상에서 조명환경을 분석하기 위한 영상처리를 수행한다. 보정된 카메라로부터 주어진 반사 구에 대한 가상의 구를 OpenGL 프레임 버퍼에 저장하여 영상에서 구 영역에 대한 마스크(mask)를 획득한다. 마스크영상을 이용하여 실제 영상에서 반사 구 영역을 잘라낸 후, 구 영역내부의 광원의 하이라이트 영역을 추출하기 위해 영상을 흑백영상으로 변환하고, 문턱값을 이용하여 어두운 영역은 제거하고 밝은 영역만 남긴다. 밝은 영역만 남은 영상을 이진화하여 광원에 대한 하이라이트 영역을 추출하고, 이에 대한 콘투어(contour)를 추출한 후 각 영역마다 색인하여 구분한다. 이때, 밝은 영역인 각각의 블롭(blob)은 반사 구상에 투영된 광원을 의미한다. 최종적으로 색인된 블롭

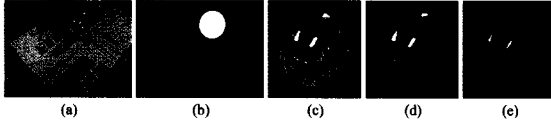


그림 7: 영상으로부터 반사 광 영역 추출; (a) 입력된 현재 영상, (b) 반사 구에 대한 마스크, (c) 마스크를 이용하여 반사 구 영역만 추출한 결과(확대), (d) 영상처리를 통해 추출한 반사 광에 대한 하이라이트 영역, (e) 콘투어 추출 결과

은 매 영상마다 계산되고 갱신되어 반사기하를 이용한 광원을 추정하는데 사용된다.

5. 반사기하 기반의 광원 추정과 생성

블롭의 콘투어 추출까지 완료가 되면, 3차원 공간에서 카메라 센터로부터 장면영상평면 상의 콘투어까지의 벡터를 알 수 있다. 카메라로부터 각 블롭의 콘투어로 향하는 광선 벡터(r_i)들은 블롭 단위로 무리(cluster)를 이루고, 한 무리의 광선벡터들은 하나의 광원으로부터 출발하여 반사구의 표면에 반사되어 카메라의 영상평면으로 사상되므로, 간단한 구면의 반사기하 원리를 이용하면 각 무리의 광원 반사 벡터를 구할 수 있다.

반사 구상에 하이라이트로 인식된 광원의 방향을 계산하기 위해서는, 카메라의 시점벡터(viewing vector)와 하이라이트가 반사된 구체 표면의 법선벡터(normal vector)를 알아야 한다. 이 벡터들은 광원 방향을 추정하는데 있어 결정적인 단서가 된다. 구체 표면($P_{surface}$)의 하이라이트 지점으로 향하는 시점벡터(\vec{r})와 반사되는 광원 벡터(\vec{l})는 아래 식 (4)과 같이 표현되며, \vec{n} 은 반사 구 상의 하이라이트 지점에서의 법선벡터를 의미한다.

$$\vec{l} = 2(\vec{n} \cdot \vec{r})\vec{n} - \vec{r}, \quad (4)$$

카메라 장면영상에서의 하이라이트 블롭은 반사 구 표면을 통해 반사되어 카메라로 투영되기때문에 영상에서의 형태가 왜곡될 수 있다. 따라서 블롭의 중앙을 광원벡터의 반사지점으로 결정하지 않고, 그림 8에서 도시한 바와 같이 블롭의 경계로부터 추출한 콘투어를 기준으로 후보 광선벡터를 선정하여 식 (4)에 의해 계산된 반사벡터들을 3차원 공간에서 각 벡터 무리(vector cluster)의 중심벡터(\vec{L} , principal vector)를 사용함으로써, 2차원 영상에서 중심값을 계산하는데 발생하는 왜곡 문제를 해결하였다. 중심벡터(\vec{L})는 $\vec{L} = (\sum_{i=1}^k \vec{l}_i) / k$ 에 의해 구하였으며, 이때, k 는 하나의 블롭을 구성하는 콘투어 픽셀의 개수이다.

중심벡터를 결정함으로써 추정된 광원방향에 대한 계산은 카메라가 촬영되는 매 프레임마다 수행된다. 하지만 카메라가 반사 구 주위를 돌거나 이동할 경우, 하이라이트가 항상 그 자리에 고정되어 나타나지 않는다. 이는 카메라가 이동할 때에

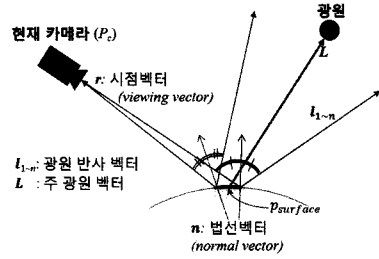


그림 8: 카메라 시점과 광원사이의 기하 관계

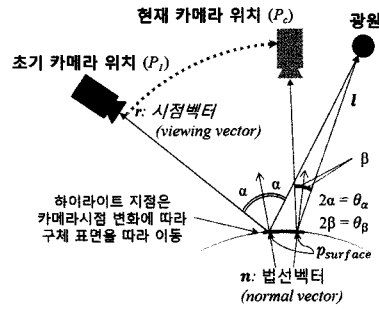


그림 9: 카메라 시점 이동에 따른 반사 기하

카메라 시점에 따른 광원 반사 지점이 구면체의 표면을 따라 이동하기 때문이다. 그림 9에 도시하였듯이 광원의 위치가 고정되어 있다고 가정하였을때, 초기 카메라 위치(P_i)에서 획득한 하이라이트 영역에 대한 콘투어는, 카메라가 이동함(P_c)에 따라 구면체 표면($P_{surface}$)을 따라 이동한다. 따라서, 실제로 3차원 공간에서의 광선벡터는 카메라 시점(CoP)에 의해 변화하므로, 모든 시점에 적용가능한, 측정 오류가 최소화 된 값을 계산할 필요가 있다.

실제 조명환경과 가장 가까운 광원방향을 추정하기 위해서는, 다양한 시점(P_i)에 대응하는 광원에 대한 중심벡터(\vec{L}_i)를 수렴하여 최소 오차의 벡터값을 계산해야한다. 반사 구를 향하는 카메라의 시점(\vec{r})이 이동하여 실제 광원이 반사 구로 향하는 벡터(\vec{l})와 가까워 졌을 때, 두 벡터 사이의 각도(θ)는 작아지고, 이때의 광원에 대한 중심벡터(\vec{L})의 값은 보다 안정적으로 계산된다. 따라서 다양한 시점에 대한 중심벡터(\vec{L}_i)들이 확보되었을 때, 시점벡터(\vec{r})와 반사되는 지점에 대한 반사 구 표면의 법선벡터(\vec{n})사이의 각도가 작아질수록 그때의 중심벡터(\vec{L})에 대한 가중치를 더 크게 부여하였고, 이는 식 (5)와 같이 표현된다.

$$\vec{L} = \sum_{i=1}^k w_i \vec{l}_i / \sum_{i=1}^k w_i, (w_i = 1/(\alpha_i + 1)), \quad (5)$$

이때, k 은 콘투어 무리의 개수이고, α_i 는 시점벡터(\vec{r})와 법선

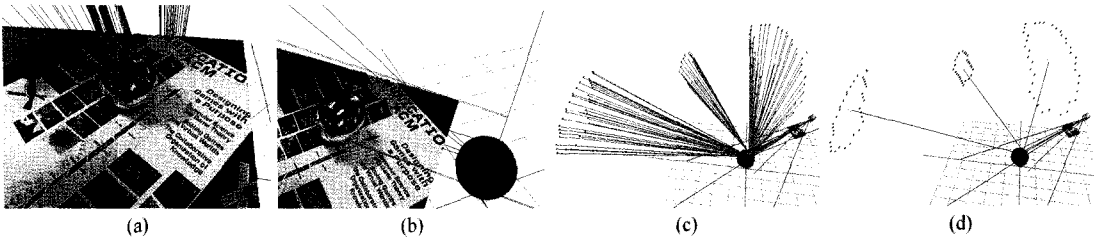


그림 10: 광원 추정을 위한 광선벡터의 3차원 가시화 결과; (a) 시점벡터(\vec{r}_i)를 초록색 선으로 표시, (b) 광원벡터(\vec{l}_i)를 붉은색 선으로 표시, (c) 광원벡터(\vec{l}_i)와 중심벡터(\vec{L}_i)를 함께 표시, (d) 중심벡터(\vec{L}_i)만 표시

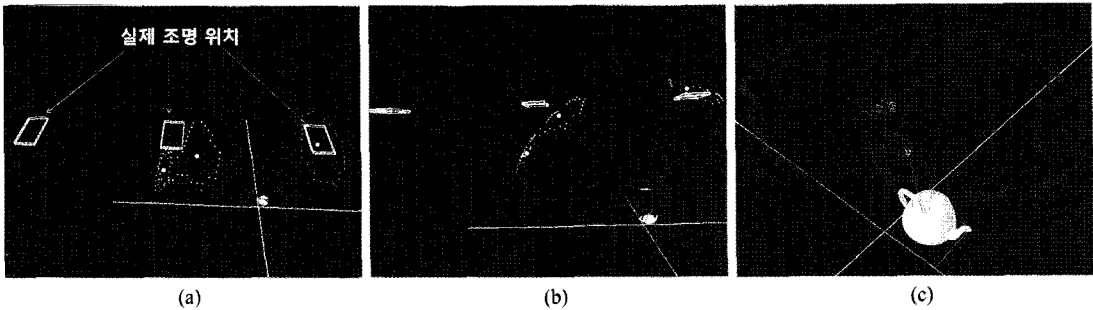


그림 11: 추정된 광원의 방향과 실제 조명위치와의 비교를 위한 3차원 가시화 결과; (a) 추정된 광원방향벡터(붉은색)를 상위 45도 각도에서 바라본 결과, (b) 추정된 광원방향벡터를 측면에서 바라본 결과, (c) 조명환경 추정을 위한 카메라의 이동 경로(초록색)

벡터(\vec{n}) 사이의 각도이다.

다중 시점에 대한 가중치를 이용하여 장면영상에 투영되는 하이라이트 영역에 대한 반사 광을 추정하는 과정은 증강현실 구동 프로그램이 수행되는 동안 연속적인 동영상으로부터 순차적으로 진행되고, 초기 위치부터 이동한 후의 현재 위치까지의 시점벡터와 반사벡터의 결과를 이용하여 현재 장면의 광원을 설정하는데 사용된다.

6. 렌더링

증강현실 환경에서 각 장면영상에 대한 광원 정보를 획득·분석하고, 계산된 조명환경을 가상객체에 적용하기 위해서는 되도록 간단하면서도 효율적으로 구동할 수 있는 방법을 이용하여 시스템을 구성하여야 한다. 반면에 가상의 객체가 실제 장면과 잘 조화되어서 정말 그럴듯하게 보이기 위해서는 많은 계산과정이 필요한 것은 피할 수 없는 문제이기도 하다.

본 논문에서는 증강현실 환경에서 실제 조명상태를 추정하는 과정과 그 결과를 렌더링하는 과정을 실시간으로 구동하기 위해서 OpenGL기반의 렌더링을 통하여 구현하였다. 이는 조명을 생성한 결과를 효과적으로 렌더링하기 위하여 필요한 조건을 만족하면서도, 시스템을 비교적 가볍게 구동하기 위함이다. 앞서 광원추정 과정을 통해 획득한 광원 방향벡터에 대

한 정보를 OpenGL의 *glLightfv*를 통하여 설정하고, OpenGL에서 읽고 렌더링할 수 있는 표준 폴리곤 모델을 사용하여 가상의 객체를 렌더링하였다. 또한, OpenGL의 스텐실 버퍼(stencil buffer)를 이용하여 그림자를 생성하고 합성하였으며, 음영효과를 위해 폰셰이딩(phong shading)으로 추정된 조명의 결과를 나타내었고, 특히 실험에 사용된 반사 구와의 비교를 위해 정반사 광(specular)을 실제와 유사하게 설정하여 본 논문에서 제안한 광원 추정 과정에 의한 가상 조명 생성의 효율성을 증명하였다.

장면에 대한 렌더링은 사용자가 카메라를 들고 반사 구를 비추며 시스템을 구동하는 시간동안 실시간으로 수행되었으며, 그 결과에 대한 스크린 샷은 그림 12, 13, 14와 같다.

7. 실험 및 결과

본 논문에서 제안한 시스템은 AMD 3.0GHz의 CPU와 nVidia GeForce GTX275 그래픽 카드를 장착한 PC에서 Windows 7 운영체제 상에서 구동되었다. 영상 획득을 위하여 Logitech Webcam Pro 9000 카메라를 사용하였다.

그림 10은 본 논문에서 제안한 카메라와 반사 구를 이용한 광원 추정 결과를 OpenGL을 이용하여 3차원으로 가시화한 결과이다. 그림 10 (a)는 카메라 센터로부터 영상평면상의 하이

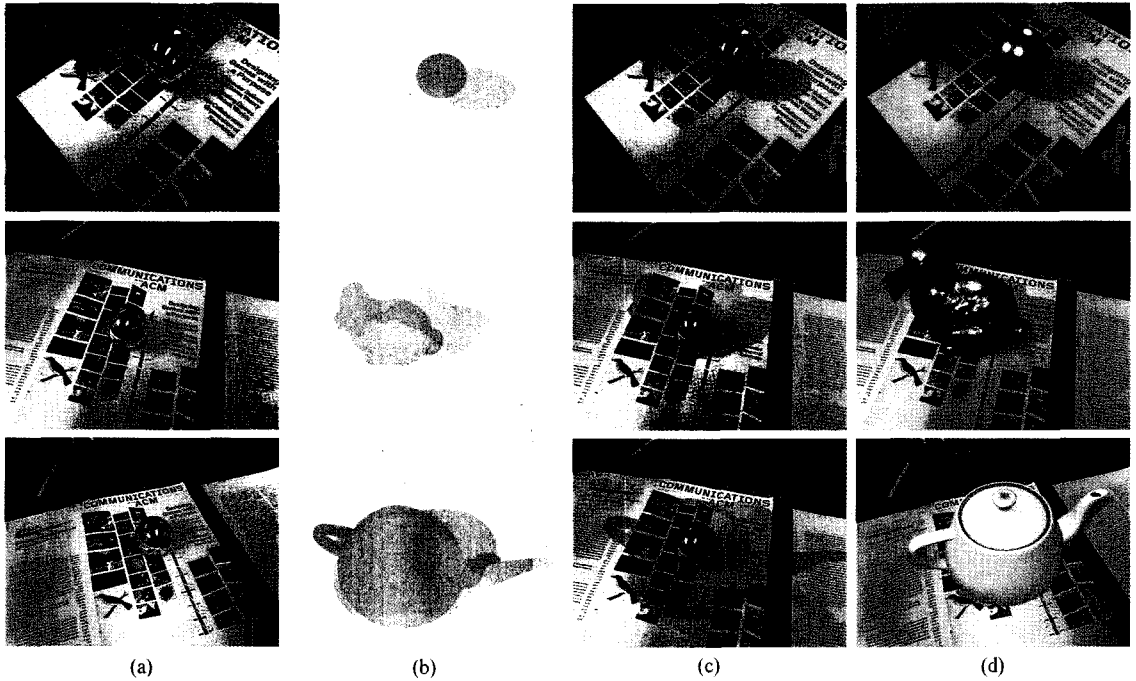


그림 12: 광원추정 결과로 가상조명을 생성한 결과; (a) 실제 영상(다중 시점 동영상 중 세 컷), (b) 가상조명으로 생성한 가상객체에 대한 그림자 마스크, (c) 그림자 합성 영상, (d) 가상 객체 증강



그림 13: 가상조명으로 생성한 가상 객체의 그림자와 반사구의 반사특성을 가상객체에 적용한 결과



그림 14: 임의의 움직이는 광원을 추정하여 가상객체에 적용한 결과(그림자, 정반사 광 표현)

라이트 영역으로 사상되는 시점벡터(\vec{r}_i)를 초록색 선으로 표시하였고, 그림 10 (b)는 영상평면(p_{image})을 통과하여 반사구 표면에 반사되는 광원벡터(\vec{l}_i)를 붉은색 선으로 표시한 결과이다. 그림 10의 (c)와 (d)는 각각, 하이라이트 콘투어에 해당하는 광원벡터(\vec{l}_i)와 중심벡터(\vec{L}_i)를 함께 표시한 결과와 중심벡터(\vec{L}_i)만 표시한 결과를 나타낸다.

그림 11은 추정된 광원의 방향과 실제 조명 위치와의 비교를 위하여 3차원으로 가시화 한 결과이다. 실제 조명의 위치는 실험이 진행된 공간에 설치된 천장 조명의 실제 위치와 크기를 측정하여 3차원 공간에 반영하였고, 같은 좌표계 상에서 실험을 수행하여 얻은 결과를 함께 표시하였다. 추정된 광원방향은 실제 환경의 조명위치와 거의 유사한 결과를 나타내었고, 실험을 진행하는 동안 카메라가 이동한 궤적은 그림 11 (c)와 같이 마커 위치에 놓여진 가상의 *Teapot* 객체 주변의 초록색 궤적으로 표시되었다.

그림 12, 13, 14에서 표시한 결과는 본 논문에서 제안한 광원 추정 방법을 이용한 가상조명 생성 기법이 증강현실 가시화에 적용 가능함을 나타낸다. 실험을 위하여 OpenGL의 기본 객체(primitive)인 *Sphere*와 *Teapot*를 이용하여 렌더링 하였으며, 추가적으로 표준 포맷(.3ds)으로 제작된 *Bunny*모델을 이용하여 몇 가지 기본적인 재질감(material)을 적용하여 구동 실험하였다.

그림 12는 본 논문에서 제안한 광원추정 기법을 통해 생성한 가상조명을 이용하여 생성된 그림자와 가상객체 렌더링 결과를 보여준다. 첫 번째 가상 객체인 구는 실험에 사용된 반사구와 같은 크기로 설정하여 원 영상에 나타난 실제 광원이 반사구에 미치는 반사광과 생성된 가상조명이 가상구에 미치는 반사광의 차이를 비교하기 위한 결과이다. *Bunny*모델과 *Teapot*은 각각 정반사광의 세기에 차이를 두어 렌더링 한 결과이고, 다른 시점으로 이동하여 관찰하더라도 환경 조명의 효과가 가상의 객체에 잘 적용됨을 알 수 있다.

그림 13은 생성된 가상조명과 함께 반사구의 반사특성을 환경매핑을 통해 가상객체에 적용한 결과이다. 반사구는 환경조명을 보정하는데 사용 될 뿐만아니라 실시간으로 주변광의 변화를 감지하여 가상객체의 렌더링에 적용할 수도 있다.

그림 14는 사용자가 임의의 움직이는 조명을 반사구 주변을 돌면서 광원 추정을 실시한 결과이다. 그림에서는 실험의 내용을 알 수 있도록 화면상에 조명을 등장시켰지만, 실제로는 조명이 화면밖에 있더라도, 반사구에 영향을 미치는 광원이라면 본 논문에서 제안한 방법으로 광원 위치 추정이 가능하다.

실험을 위해 수행한 카메라 보정, 광원 추정과 렌더링의 전 과정에 대한 수행시간은 그림 15의 그래프에서 나타내었다.

실험은 640 × 480 해상도의 카메라 영상을 실시간으로 받아들이며 수행하였으며, *Bunny*(그림 12)나 *Terrain*(그림 16)처럼 비교적 폴리곤 갯수가 많아 렌더링 시 CPU에 부하가 많이 걸리는 경우 이외에는 카메라로 촬영하며 진행되는 구동시간에 수행되었다. 수행시간에 실제로 가장 많은 비중을 차지하는

부분은 특징점 마커를 인식하고 카메라를 보정하는 과정과 폴리곤 모델을 렌더링 하는 과정인데, 그에 비해 광원의 위치를 추정하는 과정은 매우 일부에 불과하다. 본 논문에서는 증강현실 환경에서 가시적 불편함을 해소하기 위해 계산량이 많은 SIFT알고리즘을 이용하여 카메라의 자세를 보정하는 과정을 수행하였지만, 기존에 증강현실 응용프로그램에 많이 사용되는 ARToolkit [18]과 같은 마커 트래킹 라이브러리를 이용하면 현실감은 떨어지지만 더 빠른 속도를 구현할 수 있을 것이다.

본 논문에서 수행한 실험에 대한 실시간 실행 동영상은 다음 링크¹에서 확인할 수 있다.

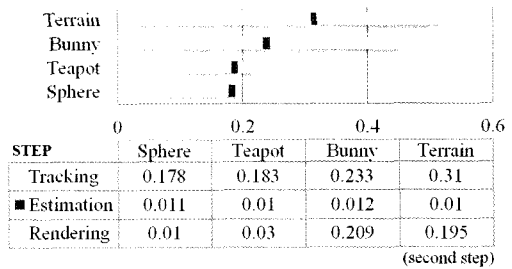


그림 15: 단계별 수행시간 평균 그래프

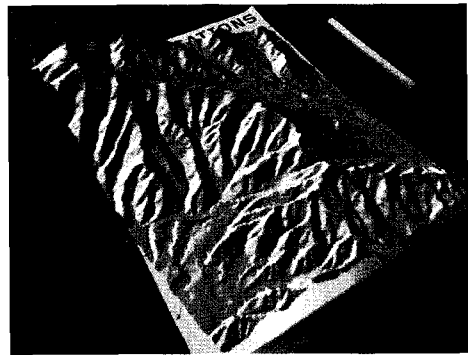


그림 16: 지형렌더링(Terrain model) 결과

8. 결론 및 향후연구

본 논문에서는 증강현실 환경에서 가상 객체를 가시화하였을 때 실제 장면과 가상 객체의 조명효과가 조화롭게 가시화될 수 있도록, 카메라 장면으로부터 실제 조명환경을 획득하여 사실적인 가상조명을 생성하는 방법을 제안하였다. 실제 조명환경을 분석하기 위해서, 반사구와 특징점 마커를 이용하여 하나의 카메라로부터 장면영상을 분석하고 실제 조명환경에 대한 조건을 분석함으로써, 일체의 사전작업이나 입력없이도

¹<http://youtu.be/P4nfzMA8XsA/>

증강현실 응용프로그램을 구동할 때 간단하게 적용하여 사용할 수 있음을 보였다.

제안하는 광원 추정 기법은 일반적인 증강현실 환경에서 쉽게 적용하여 사용할 수 있도록 실용성을 기반으로 설계되고 실험되었으므로 실시간으로 구동가능하지만, 본 논문에서 사용한 카메라 보정과정과 3차원 모델을 렌더링하는 과정에 최적화가 적용되지 않아 전체 수행시간이 비교적 길어진 것은 사실이다. 수행한 실험에서는 본 논문에서 제안한 광원 추적 기법의 정확도가 실제 환경조명을 적절히 가상환경에 적용 가능함을 나타내었지만, 향후 실용적인 사용을 위해서는 카메라 보정과 모델 렌더링을 위한 최적화가 추가적으로 수행될 필요가 있다.

감사의 글

본 연구는 2011년도 국방과학연구소의 민군겸용기술개발사업 지원과 정부(교육과학기술부) 재원으로 한국연구재단의 기초연구사업(2011-0006132)의 지원을 받아 수행되었음.

참고 문헌

- [1] F. Nathan, "A survey of light source detection methods," *Mini-Project for CMPUT 603, (Online-published)*, 2003.
- [2] Q. H. Jin-tao Ma, Ya Zhou and Y. Zhang, "Marker cube and probe sphere based illuminant detection algorithm," in *Int. Symp. Photoelectronic Detection and Imaging*, 2007.
- [3] Y. Zhang and Y.-H. Yang, "Multiple illuminant direction detection with application to image synthesis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 915–920, August 2001.
- [4] Y. Feng, "Estimation of light source environment for illumination consistency of augmented reality," in *Proc. the 2008 Congress on Image and Signal Processing*, vol. 3. IEEE Computer Society, 2008, pp. 771–775.
- [5] M. Aittala, "Inverse lighting and photorealistic rendering for augmented reality," *Visual Computer*, vol. 26, pp. 669–678, June 2010.
- [6] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *Proc. the 25th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '98. ACM, 1998, pp. 189–198.
- [7] T. Okatani and K. Deguchi, "Estimation of illumination distribution using a specular sphere," in *Proc. 15th Int. Conf. on Pattern Recognition*, vol. 3, 2000, pp. 592–595 vol.3.
- [8] M. Kanbara and N. Yokoya, "Real-time estimation of light source environment for photorealistic augmented reality," in *17th Int. Conf. on Pattern Recognition (ICPR '04)*, ser. ICPR '04, vol. 2. Washington, DC, USA: IEEE Computer Society, 2004, pp. 911–914. [Online]. Available: <http://dx.doi.org/10.1109/ICPR.2004.735>
- [9] M. H. Peter Supan, Ines Stuppacher, "Image based shadowing in real-time augmented reality," *Virtual Reality*, vol. 5, pp. 1–7, 2006.
- [10] M. Knecht, C. Traxler, O. Mattausch, W. Purgathofer, and M. Wimmer, "Differential instant radiosity for mixed reality," in *Proc. 9th IEEE Int. Symp. on Mixed and Augmented Reality (ISMAR '10)*, 2010.
- [11] M. Haller, S. Drab, and W. Hartmann, "A real-time shadow approach for an augmented reality application using shadow volumes," in *Proc. the ACM Symp. on Virtual Reality Software and Technology*, ser. VRST '03. New York, NY, USA: ACM, 2003, pp. 56–65.
- [12] M. N. Claus B. Madsen, "Towards probe-less augmented reality," in *Int. Conf. on Graphics Theory and Applications (GRAPP)*, 2008, pp. 255–261.
- [13] J. M. Frahm, K. Koeser, D. Grest, and R. Koch, "Markerless augmented reality with light source estimation for direct illumination," in *Conf. on Visual Media Production CVMP*, 2005.
- [14] S. Lee and S. K. Jung, "Estimation of illuminants for plausible lighting in augmented reality," in *Int. Symp. on Ubiquitous Virtual Reality (ISUVR '11)*, 2011.
- [15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, November 2004.
- [16] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.
- [17] 이석준, 장재석, and 정순기, "동영상 프레임간의 상호 일관성을 이용한 단일 카메라 자세 안정화 기법," in *한국HCI학회 학술대회 (HCI '11)*, 2011.
- [18] ARToolKit, *HIT Lab*. [Online]. Available: <http://www.hitl.washington.edu/artoolkit/>

〈 저 자 소 개 〉



이석준

- 2004년 대구대학교 정보통신공학부 공학사
- 2006년 경북대학교 컴퓨터공학과 공학석사
- 2008년 경북대학교 컴퓨터공학과 박사수료
- 2008년~현재 경북대학교 컴퓨터공학과 박사과정
- 관심분야 : Augmented Reality, HCI, Computer Vision 등



정순기

- 1990년 경북대학교 컴퓨터공학과 공학사
- 1992년 한국과학기술원 전산학과 이학석사
- 1997년 한국과학기술원 전산학과 공학박사
- 1998년~현재 경북대학교 IT대학 컴퓨터학부 교수
- 관심분야 : Virtual Reality, Artificial Interlligence, Computer Vision, Imge Processing, Computer Graphics.