



라즈베리 파이 클러스터를 이용한 4D 라이트 필드 카메라 시스템 개발

Development of 4D Light Field Camera Systems Using Raspberry Pi Clusters

저자
(Authors) 안재풍, 서웅, 김영욱, 박상훈, 임인성
Jaepung An, Woong Seo, Youngwook Kim, Sanghun Park, Insung Ihm

출처
(Source) [한국정보과학회 학술발표논문집](#), 2015.12, 1110-1112 (3 pages)

발행처
(Publisher) [한국정보과학회](#)
KOREA INFORMATION SCIENCE SOCIETY

URL <http://www.dbpia.co.kr/Article/NODE06602642>

APA Style 안재풍, 서웅, 김영욱, 박상훈, 임인성 (2015). 라즈베리 파이 클러스터를 이용한 4D 라이트 필드 카메라 시스템 개발. 한국정보과학회 학술발표논문집, 1110-1112.

이용정보
(Accessed) 송실대학교
114.71.40.***
2016/11/26 04:27 (KST)

저작권 안내

DBpia에서 제공되는 모든 저작물의 저작권은 원저작자에게 있으며, 누리미디어는 각 저작물의 내용을 보증하거나 책임을 지지 않습니다. 그리고 DBpia에서 제공되는 저작물은 DBpia와 구독계약을 체결한 기관소속 이용자 혹은 해당 저작물의 개별 구매자가 비영리적으로만 이용할 수 있습니다. 그러므로 이에 위반하여 DBpia에서 제공되는 저작물을 복제, 전송 등의 방법으로 무단 이용하는 경우 관련 법령에 따라 민, 형사상의 책임을 질 수 있습니다.

Copyright Information

Copyright of all literary works provided by DBpia belongs to the copyright holder(s) and Nurimedia does not guarantee contents of the literary work or assume responsibility for the same. In addition, the literary works provided by DBpia may only be used by the users affiliated to the institutions which executed a subscription agreement with DBpia or the individual purchasers of the literary work(s) for non-commercial purposes. Therefore, any person who illegally uses the literary works provided by DBpia by means of reproduction or transmission shall assume civil and criminal responsibility according to applicable laws and regulations.

라즈베리 파이 클러스터를 이용한 4D 라이트 필드 카메라 시스템 개발

안재풍^{0,1} 서웅¹ 김영욱¹ 박상훈² 임인성¹

서강대학교 공과대학 컴퓨터공학과¹ 동국대학교 영상대학원 멀티미디어학과²
{ajp5050,wng0620,kimyu7,ihm}@sogang.ac.kr mshpark@dongguk.ac.kr

Development of 4D Light Field Camera Systems Using Raspberry Pi Clusters

Jaepung An^{0,1} Woong Seo¹ Youngwook Kim¹ Sanghun Park² Insung Ihm¹

Dept. of Comp. Sci. & Eng. Sogang University¹ Dept. of Multimedia Dongguk University²

요 약

3차원 공간에서 임의의 방향으로 진행하는 빛을 수학적으로 기술하는 4D 라이트 필드에 대한 개념이 컴퓨터 그래픽스 분야에서 소개되었다. 실세계에서 이 개념을 유용하게 활용하기 위해서는 라이트 필드 데이터 촬영용 하드웨어 장비와 응용 소프트웨어의 개발이 필수적이다. 최근 출시된 상용 카메라는 라이트 필드 데이터의 응용 범위 조절이 제한적이기 때문에 실제 활용도가 낮고, 일반인이 응용 상황에 적합한 고성능의 라이트 필드 카메라를 직접 제작하기에는 한계가 있다. 본 논문에서는 라즈베리 파이 클러스터를 이용하여 적은 비용으로 원하는 수준의 라이트 필드 데이터를 촬영할 수 있는 확장 가능하고 유연한 구조를 갖는 카메라 시스템과 응용 프로그램의 설계 및 구현 과정에 대해 설명한다. 사용자는 개발된 시스템을 이용하여 원하는 규격의 라이트 필드 데이터를 촬영할 수 있고, 생성된 데이터에 대한 원근시점 이동, 재초점, 피사계 심도 조절과 같은 유용한 기능을 자유롭게 제어할 수 있다. 본 논문에서 제시하는 시스템은 향후 확장성과 유연성을 지원하는 새로운 형태의 모듈화된 라이트 필드 카메라의 개발에 핵심적인 기술로 활용될 수 있을 것이다.

1. 서론

컴퓨터 그래픽스 분야에서 90년대 중반 4D 라이트 필드 렌더링(light field rendering) 이론이 소개된 이래로[1], 최근에는 일반 사용자들을 위한 상용 라이트 필드 카메라가 판매되고 있다[2,3]. 전통적인 라이트 필드 렌더링은 카메라 평면에 동일한 간격으로 배치된 복수의 카메라에서 초점면(focal plane)을 촬영하여 생성된 라이트 필드 데이터의 접근을 기반으로 한다. 라이트 필드 데이터 $L(u,v,s,t)$ 는 카메라의 기준점 좌표 (u,v) 와 초점면의 샘플링 위치 좌표 (s,t) 를 인자로 갖는 4D 함수이다. 사용자는 제공되는 응용 프로그램으로 라이트 필드 영상에 대해 원근시점 이동(perspective view changes), 재초점(refocusing)[4], 피사계 심도 조절(depth of field controls)과 같이 디스플레이에 직접적인 영향을 끼치는 효과들을 실시간 제어할 수 있다. 하지만 현재 판매되고 있는 상용 카메라의 경우, 하드웨어의 구조적 한계로 이런 효과들은 매우 제한적인 범위에서만 조절 가능하며 이로 인해 실제 카메라의 활용도는 낮은 수준이다. 따라서 이런 유용한 기능의 활용 수준을 사용자의 기대에 맞추기 위해서는 카메라 개수, 배치, 렌즈 종류 등을 필요에 따라 자유롭게 선택할 수 있는 라이트 필드 카메라 시스템이 요구된다. 뿐만 아니라 확장가능하고 유연한 구조를 갖도록 설계된 촬영 시스템과 연동하여 직관적인 영상처리 작업 환경을 제공하는 응용 프로그램도 반드시 필요하다. 본 논문에서 소개하는 라즈베리 파이(Raspberry Pi) 클러스터 기반의 라이트 필드 카메라 시스템은 이러한 요구를 만족시킬 수 있도록 구현되었다.

개발된 라이트 필드 카메라 시스템은 상용 카메라와 비교했을 때 다음과 같은 특징을 갖는다. 우선 카메라의 구성과 배치, 렌즈의 선택이 자유롭게 때문에 촬영된 영상의 개수, 해상도, 광각(field of view)을 필요에 따라 조정할 수 있는 확장성과 유연성을 제공한다. 또한 사용된 라즈베리 파이와 카메라의 가격이 낮은 수준이라 비교적 저가로 시스템을 제작할 수 있다. 하지만 개발 과정과 실제 사용을 위해서 전문적인 지식이 필요하고 이동의 편의성 측면에서 약점을 갖는다. 그러나 정밀한 설정이 필요한 부분을 효율적으로 모듈화 한다면 작업 비용을 최소

화 하면서 사용자가 원하는 수준과 구조를 갖는 고성능 라이트 필드 카메라를 활용할 수 있을 것이다.

2. 관련연구

Wilburn 등[5]은 카메라 어레이(array)를 이용해 연구용 라이트 필드 카메라 시스템을 구축하는 방법을 제안하였다. 카메라의 배치가 자유롭기 때문에 라이트 필드의 범위를 자유롭게 조절할 수 있고, 상용 카메라에 따라 고품질의 데이터를 얻어낼 수 있다. 하지만 틀은 물론 보드 또한 직접 제작해야 하는 번거로움이 있고, 이로 인해 제작비용이 높고 사용이 쉽지 않다는 단점이 있다. 또한 여러 대의 PC를 사용하기 때문에 휴대가 매우 불편하다. Ng 등[4]은 휴대용 카메라에 마이크로 렌즈를 장착한 휴대용 라이트 필드 카메라를 제작하였다. 하지만 마이크로 렌즈의 제작과 이를 카메라에 장착하는 과정이 복잡하기 때문에 일반 사용자들이 이용하기에는 어렵고 제작비용이 높다. 또한 휴대용 카메라의 크기로 인해 라이트 필드의 범위가 제한된다는 단점이 있다. 그 밖의 논문[6]을 통해 발표된 연구용 카메라 시스템의 경우도 위의 두 가지 방법과 비교해서 큰 차이가 없다.

대표적인 상용 제품으로는 리트로(Lytro) 카메라[2]와 레이트릭스(Raytrix) 카메라[3]가 있다. 리트로 카메라의 모델들은 앞에서 설명한 Ng 등이 제안한 방법을 기반으로 생산되었다. 가격이 낮고 다루기 쉬우며 휴대가 편리하지만, 촬영 영상 정보가 적어 라이트 필드 데이터를 활용한 유용한 효과들이 잘 표현되지 않는다. 레이트릭스 카메라의 제품들도 성능 면에서 리트로 카메라와 유사하지만 상대적으로 가격이 비싸고 카메라 외에 본체가 없으면 구동하지 못하기 때문에 휴대성이 떨어진다.

3. 하드웨어 설계 및 구현

본 연구의 시스템은 크게 본체 부분과 카메라 부분으로 나누어 설계하였다(그림 1 참조). 본체 부분은 호스트 PC와 라즈베리 파이 본체, 공유기, 그리고 전원 부분으로 구성된다. 사용자는 호스트 PC를 통해

명령을 입력하고 이것은 개별 라즈베리 파이 본체로 전달되어 라이트 필드 카메라 시스템을 제어한다. 라즈베리 파이 본체는 공유기와 연결되며 이를 통해 호스트로부터 받은 명령이 각 라즈베리 파이에서 병렬적으로 수행되도록 구성하였다. 라즈베리 파이의 주된 역할은 영상을 촬영하고 전송하는 일이기 때문에 라즈베리 파이 간에 통신이 불필요하고 따라서 자연스럽게 병렬화가 가능하다.

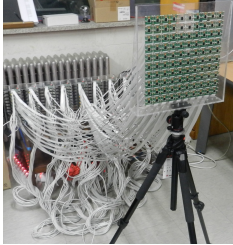


그림 1: 시스템 구조

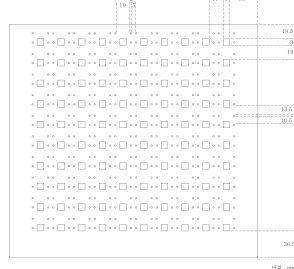


그림 2: 카메라 어레이 배치

카메라 부분은 라즈베리 파이 본체와 연결된 카메라 모듈, 이들을 고정하는 틀, 그리고 삼각대로 구성된다. 각 카메라는 개별적으로 작동하기 때문에 이들을 고정하는 틀과 카메라의 배치에 따라서 라이트 필드 데이터의 규칙이 결정된다. 본 연구에서는 10×10 , 총 100개의 카메라 모듈로 구성된 카메라 어레이를 제작하였고, 이 부분은 라이트 필드 카메라 시스템의 렌즈에 해당한다(그림 2의 설계도 참조). 카메라 사이 간격을 $28mm$ 로 설정하였기 때문에 전체 카메라 어레이 영역의 크기는 $280mm \times 280mm$ 가 된다. 개별 카메라를 구별하기 위해 카메라 어레이에서 (i, j) 에 위치하는 카메라에 $i \times 10 + j$ 의 고유번호를 부여했다.

라이트 필드 카메라 시스템을 효과적으로 사용할 수 있도록 각 라즈베리 파이마다 내부 IP 주소를 할당하였고, 이를 통해 개별적으로 SSH로 명령을 내릴 수 있다. 현재 시스템은 호스트 PC에서 촬영 명령을 내리면 카메라 모듈이 두 개의 그룹으로 50개씩 묶여 병렬적으로 촬영을 진행한다. 각 카메라의 촬영 결과는 화질 100%인 JPEG 형식으로 저장되며, 하나의 크기가 약 2.7Mbytes이므로 동시에 촬영된 100개의 영상 전체는 약 270Mbytes가 된다.

4. 소프트웨어 설계 및 구현

4.1. 가상 카메라 모델

그림 3은 본 연구에서 개발한 라이트 필드 카메라의 개념적 구성을 보여준다. 뷰 윈도우(view window)는 각 카메라 모듈이 세상을 들여다보는 창문에 해당하며 이를 기준으로 세상 좌표계(world coordinate, WC)가 정의된다. 실제 촬영 시에는 현실 공간에서 뷰 윈도우 영역을 정하기 위하여 체크 보드를 사용하여 보드 상의 특정 사각형 영역이 뷰 윈도우 영역으로 매핑되도록 설정한다. 카메라 어레이는 가상 렌즈 평면을 구성한다. 실제 촬영 시 가급적 이 평면과 뷰 윈도우가 평행하고 가상 렌즈 중심과 뷰 윈도우의 중심이 일치하도록 배치하지만 오차는 존재할 수밖에 없다. 세상 좌표계에서 가상 렌즈 평면 좌표계는 평면 상의 왼쪽 아래 모서리 p 와 직교기저 q_u, q_v, q_n 으로 정의된다. 이후 캘리브레이션 과정에서 외부행렬 추정 계산을 통해 각 카메라 모듈에 대한 세상 좌표계에서의 투영 참조점(projection reference point)이 결정된다. 각 카메라의 투영 참조점은 가상 렌즈 평면을 이산적으로 샘플링하는 빛 포집 지점(light sampling point)이 된다. 개별 카메라가 촬영하여 산출한 원시 데이터 간에는 상호 연결관계가 존재하지 않으므로 원시 데이터로부터 평면 호모그래피(planar homograph) 계산을 통하여 뷰 윈도우 영역을 균일하게 샘플링 하는 영상을 생성할 수 있다. 캘리브레이션 후 각 카메라가 촬영한 영상은 $[0,1] \times [0,1]$ 로 정의되는 정규좌표 공간에서의 4D 라이트 필드를 구성한다(그림 3의 좌상단의 그림 참조). 함수 $L(u, v, s, t)$ 는 뷰 윈도우의 (s, t) 지점을 통과하여 가상 렌즈 평면의 (u, v) 지점으로 들어오는 빛의 광휘를 의미하게 된다.

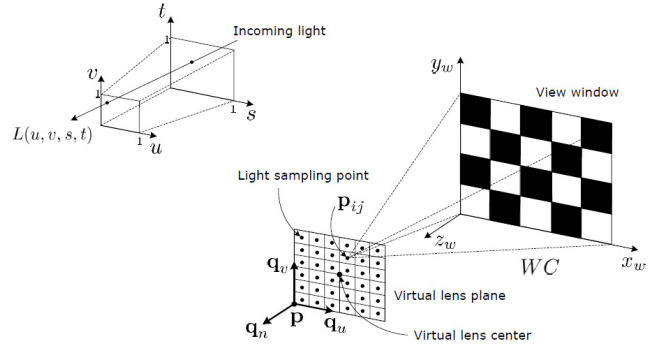


그림 3: 개발 라이트 필드 카메라 시스템의 개념적 구성도

4.2. 렌더링 프로세스

사용자가 조작할 수 있는 가상 카메라 프레임은 투영 참조점 e_c 와 직교 단위 벡터 u_c, v_c, n_c 로 정의된다. 중심이 n_c 축에 정렬되도록 가상 필름을 배치하면 임의의 필름 샘플링 지점에는 카메라 프레임의 원점이 되는 e_c 를 통과하는 투영선을 따라 들어오는 빛이 상을 맺게 된다.

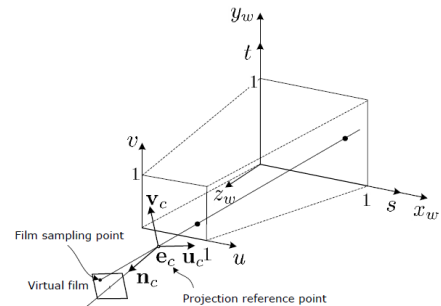


그림 4: 세상에 대한 렌더링 카메라 배치

그림 4는 가상 카메라를 4D 라이트 필드 데이터가 존재하는 세상 좌표계에 배치한 상황을 보여준다. 사용자는 마우스와 키보드를 사용하여 가상의 카메라 위치와 방향을 자유롭게 제어할 수 있다. 사용자의 조작을 통해 임의로 설정된 가상 카메라로 촬영될 영상을 라이트 필드 데이터로부터 계산하는 것을 원근시점 이동이라 한다. 이 기법은 가상의 카메라에서 광선을 쏘아 라이트 필드 데이터와의 교점을 찾는 연산을 기본으로 하는데, 이 과정은 사실적 렌더링 알고리즘 가운데 하나인 광선 투사법(ray casting)과 유사하다. 결과 이미지 상의 픽셀 (i, j) 로 투영되는 광선의 방향 벡터 r_{ij} 는 화면 종횡비 a , y 축에 대한 관측시야 각도 θ 가 주어졌을 때 해상도 1024×1024 에 대해 다음과 같이 계산된다:

$$r_{ij} = -n_c + a \sin \theta \left(\frac{2i}{1024} - 1 \right) u_c + \sin \theta \left(\frac{2j}{1024} - 1 \right) v_c$$

가상의 카메라 위치가 P_{vc} 일 때, 매개 직선 $R(t) = P_{vc} + tr_{ij}$ 와 라이트 필드의 두 평면 uv, st 와의 교점을 각각 찾은 후, 그에 대응하는 라이트 필드 데이터 값을 액세스하고 이를 보간하여 픽셀 (i, j) 의 컬러를 구할 수 있다. 사용자가 카메라 위치와 방향을 바꿀 때 마다 모든 픽셀에 대해 이를 반복 수행하여 실시간 원근시점 이동 효과를 구현하였다.

Ng[4]가 소개한 라이트 필드 카메라 재초점의 기본 원리를 기반으로 본 시스템에서도 이 기능을 구현하였다. 재초점 계산을 위해서는 가상 렌즈를 정의하고 이를 이용하는 가상 카메라를 구현해야 한다. 시스템을 구성하는 전체 카메라들을 가상 렌즈라고 가정하면 카메라가 배치된 평면을 가상 렌즈가 위치한 가상 렌즈 평면으로 생각할 수 있다(그림 5 참조). 재초점 기능을 위해서는 가상 렌즈 평면과 초점면 사이의 거리인 초점 거리를 조절할 수 있어야 한다. 초점 거리를 수정하면 가상 렌즈 평면의 상이 맺히는 가상 영상 평면의 위치도 변화된다. 가상 영상 평면과 가상 렌즈 평면 사이의 거리 d_l , 가상 렌즈 평면과 물체 사이의

거리 d_2 , 초점 거리 f 사이에는 $\frac{1}{d_1} + \frac{1}{d_2} = \frac{1}{f}$ 의 관계가 성립한다. 여기서 초점 거리 f 를 수정하면 d_1 이 바뀌고 가상 영상 평면의 위치가 변경되면서 영상 영역의 크기도 변화된다. 가상 렌즈 중심을 지나가는 빛은 굴절하지 않는다는 점을 이용하여 수정된 초점 거리 f' 에 의해 생성될 영상의 광휘는 다음과 같이 구할 수 있다:

$$E_f(i, j) = \frac{1}{100\alpha^2 f^2} \sum_{v=0}^9 \sum_{u=0}^9 L_f(u, v, u(1 - \frac{1}{\alpha}) + \frac{s}{\alpha}, v(1 - \frac{1}{\alpha}) + \frac{t}{\alpha})$$

여기서 $\alpha = f'/f$ 이고, (s, t) 는 픽셀 (i, j) 와 가상 렌즈 중심을 관통한 광선이 새로운 초점면과 만나는 교점이다.

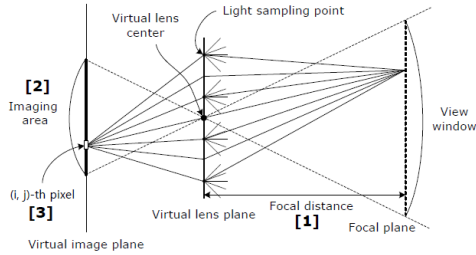


그림 5: 재초점 방법

앞에서 설명한 가상 카메라는 렌즈의 면적이 0인 바늘구멍 카메라 모델을 기본으로 설계 되었다. 하지만 재초점이나 피사계심도(depth of field)등의 기능을 구현하기 위해서는 실제 카메라처럼 면적을 갖는 렌즈가 고려되어야 한다. 이를 위해 그림 6과 같이 두 벡터 u_c, v_c 로 결정되는 평면 위에 e_c 가 중심이 되도록 배치된 새로운 원형의 가상 렌즈를 정의한다. 사용자는 이렇게 정의된 가상 렌즈의 구경(aperture)을 변경함으로써 자유롭게 피사계심도 효과를 조절할 수 있다.

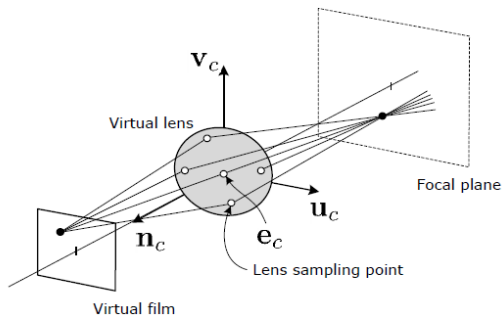


그림 6: 가상 렌즈를 갖는 가상 카메라 구조

실제 구현에서는 무한 샘플링이 불가능하므로 유한 렌즈 샘플링 지점에서 샘플링을 수행해야 한다. 유한개의 렌즈 샘플링 지점은 카메라의 위치에 관계없이 결정된다. 렌즈 샘플링 지점 결정 방법으로 특징한 템플릿을 사용한 배치와 난수를 이용한 기법을 생각할 수 있다. 전자는 후자에 비해 안정적이지만 샘플링 위치의 고정으로 가상 렌즈로 들어오는 빛을 자연스럽게 표현하기에 한계를 나타낸다. 후자에서는 난수를 이용하여 균등한 확률로 렌즈 샘플링 지점을 결정한다.

4. 실험결과

본 절에서 설명할 성능 평가 실험을 위해 Intel Xeon X5690 CPU, Nvidia Geforce GTX 690 GPU, 12GB 메인 메모리가 탑재된 PC를 사용하였다. 원근시점 이동 기능의 속도 향상을 위해 OpenCL 1.1 API를 사용하였으며, GPU 가속 부분은 원근시점 이동을 위해 가상 카메라에서 광선을 쏘는 부분이다. 원근시점 이동은 가상 카메라가 촬영하는 영상의 해상도에 비례해서 계산량이 증가하고, 가속 효과로 CPU 1 쓰레드만 사용한 경우에 비해 약 1,000배 정도 속도가 향상되는 것을 확인

할 수 있다. 해상도 1024×1024 영상의 원근시점 이동에 걸리는 시간은 약 1.6msec 이다. 그림 7은 가상 카메라 위치를 카메라 평면의 중앙에 고정한 상태에서 회전을 적용한 결과이다.

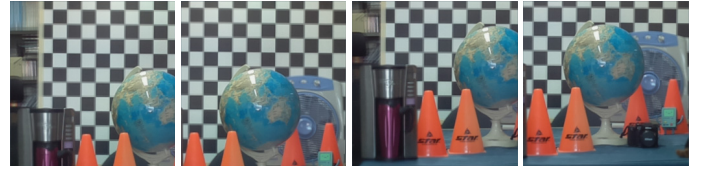


그림 7: 원근시점 회전 적용 결과

재초점 기법 실험은 크기가 $280mm \times 280mm$ 인 10×10 카메라 어레이를 하나의 가상 렌즈로 고려하고, 초점거리를 변화하였을 때 변화된 빛의 이동 방향을 계산해서 가상 렌즈로 촬영한 영상을 생성하는 과정이다. 실제 카메라의 위치는 가상 렌즈에 대한 샘플링 지점으로 생각할 수 있고, 실제 카메라에 비해 가상 렌즈의 직경이 훨씬 커서 피사계심도가 얇다고 볼 수 있기 때문에 재초점 결과가 잘 표현될 수 있다. 원근시점 이동 실험과 마찬가지로 가속 부분은 가상 카메라에서 광선을 쏘는 부분이며 해상도 1024×1024 영상의 재초점에 걸리는 시간은 약 19msec 이고, 그림 8은 재초점 실험 결과 영상이다.

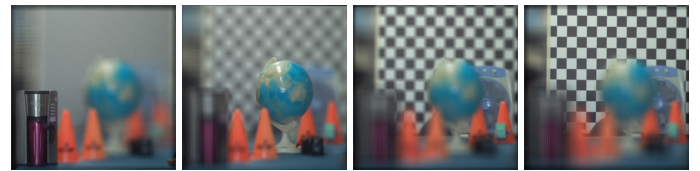


그림 8: 재초점 기법 적용 결과

5. 결론

본 논문에서는 자유로운 카메라 배치를 통해 라이트 필드 데이터 규격에 대한 제약을 최소화 할 수 있는 유연하고 확장 가능한 라이트 필드 카메라 시스템의 설계와 제작에 대해 설명하였다. 사용자는 재구성 가능한 라즈베리 파이 클러스터 모듈을 이용해 본인의 목적과 응용에 맞는 적절한 구조로 시스템을 구성함으로써 원하는 규격의 라이트 필드 데이터를 촬영할 수 있다. 그리고 GPU 가속 기능을 활용하도록 구현된 실시간 응용 프로그램을 이용해 기대하는 영상 효과들을 정밀하게 제어할 수 있다. 특히 본 시스템은 광선 추적 기반의 알고리즘 구현을 통해 기존의 원근시점 이동, 재초점, 피사계 심도 조절 기능을 개선하였으며 개별 효과를 동시에 적용할 수 있는 진보된 기능을 제공한다.

감사의 글 "본 연구는 미래창조과학부 및 정보통신기술진흥센터의 ICT/SW 창의과정(SW중심대학) 지원사업의 연구결과로 수행되었음" (R0215-15-1001, R2215-15-1003)

참고문헌

- [1] M. Levoy and P. Hanrahan, "Light field rendering," in Proceedings of SIGGRAPH '96, pp. 31-42, ACM, 1996.
- [2] Lytro, "Lytro." <https://www.lytro.com/camera/specs/>.
- [3] Raytrix, "Raytrix." <https://raytrix.de/index.php/Cameras.html>.
- [4] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," Computer Science Technical Report CSTR, vol. 2, 2005.
- [5] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," ACM Trans. Graph., vol. 24, pp. 765-776, July 2005.
- [6] T. Georgeiv, K. C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala, "Spatio-angular resolution tradeoffs in integral photography," in Proceedings of EGSR '06, pp. 263-272, 2006.