

초박형 라이트필드 카메라의 실시간 분해능 향상 알고리즘 개발

김동건^{1o}

유재관²

조용진²

김민혁^{1*}

¹한국과학기술원

²LIG 넥스원 주식회사

¹{dgkim, minhkim}@vclab.kaist.ac.kr

²{jaekwan.ryu, yongjin.jo}@lignex1.com

Improving Spatial Resolution in Real-time for Ultra-thin Light Field Cameras

Donggun Kim^{1o}

Jaekwan Ryu²

Yongjin Jo²

Min H. Kim^{1*}

¹KAIST

²LIG Nex1 Co., Ltd

요약

초박형 라이트필드 카메라 시스템은 이미지 센서 위에 렌즈 어레이를 부착하는 방식으로 만들어진다. 이러한 초박형 라이트 필드 카메라는 하나의 이미지 센서를 여러 개의 sub-aperture가 나눠쓰는 방식으로 되어있어 개별 이미지의 분해능이 낮으며, sub-aperture 이미지들을 융합해 추가적인 분해능 향상이 수행되어야 한다. 본 연구에서는 초박형 라이트필드 카메라 시스템을 개발했으며, 개발된 카메라 시스템을 위한 실시간 분해능 향상 알고리즘을 개발, 실험을 통해 검증했다. 개발된 초박형 라이트필드 카메라는 두께 2mm, 24개(6×4)의 551×551 해상도의 sub-aperture로 구성되어 있으며, 임베디드 컴퓨팅 보드를 사용해 휴대가 가능하도록 제작되었다. 실시간 분해능 향상 알고리즘은 임베디드 컴퓨팅 보드의 GPU에서 병렬처리를 통해 라플라시안 피라미드 기반의 이미지 융합 알고리즘을 수행한다. 실험을 통해 검증한 결과로, 개발 시스템은 MTF50값이 평균 35% 정도 개선되었으며, 10.65fps의 처리속도로 실시간 처리가 가능함을 확인했다.

키워드: 라이트필드 카메라, 다중 시점 카메라, 분해능 향상, 실시간 융합

Keywords: light field camera, multi-view camera, spatial resolution improvement, real-time fusion

1. 서론

라이트필드(Light field)[1, 2] 이미징은 한번의 촬영으로 촬영 대상의 빛의 세기만을 얻을 수 있는 기존 이미징 기술과는 다르게, 한번의 촬영으로 대상의 빛의 세기 뿐만 아니라 방향까지도 획득할 수 있는 기술이다. 한번의 촬영을 통해 빛의 세기와 방향을 동시에 획득할 수 있다는 점에서 라이트 필드는 촬영된 이미지의 re-focusing, 깊이 정보 획득, 고해상도 영상 복원, view synthesis 등 다양한 분야에서 사용될 수 있다.

라이트필드 카메라는 갠트리(gantry)에 여러개의 개별 카메라를 부착하는 방식[3]과 일반적인 카메라와 같은 크기에 마이크로 렌즈 어레이를 이용해 여러 방향에서 오는 빛을 촬영하는 방식등의 방식이 있으나, 최근 휴대성을 고려한 후자의 방식으로 많은 연구가 진행되고 있다. 하지만 두가지 경우 모두 카메라의 소형화에는 한계가 있으며, 물리적으로 라이트필드와 같은 다중시점 카메라를 소형화하기 위한 연구 또한 진행되고 있다.

초박형 라이트필드 카메라는 앞서 말한 소형화된 라이트필드 카메라의 한 종류로, 이미지 센서 앞에 렌즈어레이를 부착한 형태로 제작된다[4]. 각각의 렌즈어레이는 이미지 센서에 하나의 영상을 만들어내고, 각각의 영상을 sub-aperture 이미지라 한다. 이러한 초박형 라이트필드 카메라의 경우 하나의 이미지 센서를 여러 sub-aperture가 나눠 사용하기 때문에 개별 sub-aperture들의 분해능이 낮아 영상 융합을 통한 분해능 향상이 필요하다.

본 연구에서 대상으로 하는 환경은 50cm 이상의 거리를 대상으로, 최종적으로는 무인체 등과 같이 휴대성이 강조되는 사용처에 탑재되어 실시간으로 영상 처리를 가능하게 하는 것이 목표이다. 이를 가능하게 하기 위해 휴대가 가능한 초박형 라이트필드 카메라 시스템을 제작했으며, 실시간으로 고분해능 영상을 획득하는 방법을 개발해 실험을 통해 검증했다.

*corresponding author: Min H. Kim/KAIST(minhkim@vclab.kaist.ac.kr)

2. 초박형 라이트필드 카메라 시스템

2.1 카메라 하드웨어 구성

본 연구에서 사용된 초박형 라이트필드 카메라는 IMX477 이미지 센서 위에 자체 제작한 렌즈 어레이를 장착해 만들어졌다. (Figure 1 좌측) 부착된 렌즈 어레이의 경우 두께는 약 0.5mm, 전체 초박형 라이트필드 카메라의 두께는 약 2mm 이며, 렌즈의 초점이 이미지 센서에 맞게 하기 위해 적절한 간격을 두고 부착되었다. 최종적으로 사용 가능한 sub-aperture는 가로 6개 × 세로 4개, 각각 551×551 해상도의 흑백 이미지들로 구성되어 있다. 초박형 라이트 필드 카메라를 통해 획득된 이미지는 Figure 2 에서 확인할 수 있다.

카메라 시스템은 무인체 탑재를 위한 휴대성과 실시간 이미지 처리에 필요한 성능을 고려해 임베디드 컴퓨팅 보드인 Nvidia Jetson NX 기반으로 제작되었으며, 임베디드 보드에 내장된 MIPI-CSI2 인터페이스를 통해 카메라와 통신할 수 있다. Figure 1 에서 초박형 라이트필드 카메라 및 임베디드 컴퓨팅 보드와 케이스가 포함된 카메라 시스템의 모습을 확인할 수 있다.

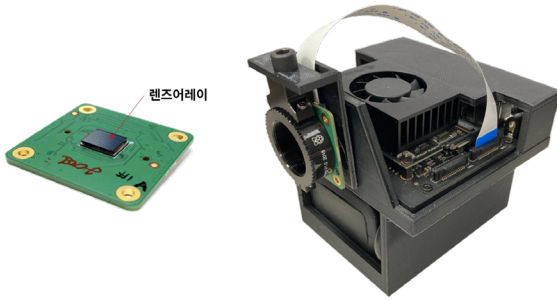


Figure 1: Ultra-thin light field camera with lens array (left), Developed camera system(right).

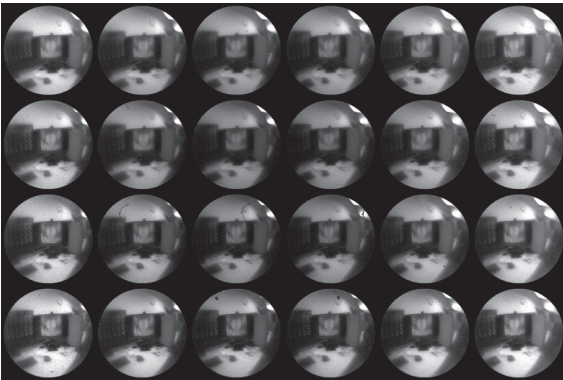


Figure 2: Single-shot image from ultra-thin light field camera.

2.2 카메라 캘리브레이션

제작된 초박형 라이트필드 카메라를 사용하기 위해 우선 렌즈셰이딩(lens shading) 보정이 수행됐다. 우선 깨끗한 무광 백색 표

면을 촬영해 렌즈의 중심부분 대비 가장자리에서 발생하는 명도 감소를 측정하고, 이를 이용해 보정하기 위해 곱해져야 하는 값을 계산해 저장했다. 이후 실제 촬영 시에는 사전에 계산된 렌즈셰이딩 보정값을 이용해 이미지에 보정값을 곱해줌으로써 보정을 수행했다[5].

또한, 일반적으로 렌즈에서 발생하는 기하학적 왜곡(geometric distortion) 보정 또한 수행 되었다. 이를 위해 아래 수식 (1) 의 계수가 2개인 radial distortion과 tangential distortion을 고려한 모델을 기반으로 체크보드 이미지 촬영과 최적화를 통해 distortion model의 계수를 알아내고, 이를 통해 이미지의 왜곡을 보정했다.

$$\begin{aligned} r^2 &= x^2 + y^2, \\ x' &= x(1 + k_1r^2 + k_2r^4) + (2p_1xy + p_2(r^2 + 2x^2)), \\ y' &= y(1 + k_1r^2 + k_2r^4) + (p_1(r^2 + 2y^2) + 2p_2xy). \end{aligned} \quad (1)$$

여기서 x 와 y 는 이미지 중심이 원점인 좌표계에서 가로, 세로 픽셀 좌표이며, x' , y' 은 왜곡 모델에 의해 얻어지는 좌표이다. 렌즈 셰이딩 보정과 기하학적 왜곡 보정을 적용하기 전, 후의 결과는 Figure 3에서 확인 할 수 있다.

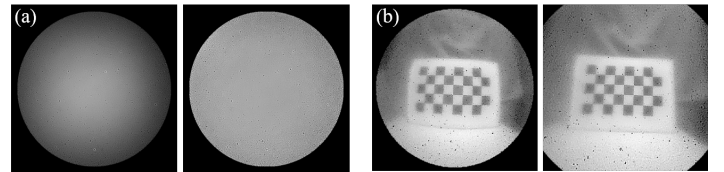


Figure 3: (a) lens shading correction: before (left), after (right), (b) Distortion correction: before (left), after (right).

3. 실시간 분해능 향상 알고리즘

촬영된 여러 이미지를 실시간으로 융합해 분해능이 향상된 이미지를 획득하는 알고리즘은 이미지를 정렬하고, 정렬된 이미지를 융합하는 두 단계로 되어있다.

3.1 이미지 정렬

일반적인 라이트필드 카메라의 경우 sub-aperture 이미지들의 baseline 차이로 인해 촬영 대상의 깊이(depth)에 따라 시차(parallax)가 발생하게 된다. 그러나, 초박형 라이트필드 카메라의 경우 짧은 초점거리로 인해 카메라로부터 일정 거리 이상에있는 대상의 경우 sub-aperture 이미지들은 시차가 0에 가까워지는 특성과, imaging plane이 하나의 센서라는 점을 고려하면 단순 평행이동(translation)을 통해 정렬이 가능하다. 이를 위해 우선 원거리에 있는 대상을 다수 촬영하고, ORB[6] 특징점을 통해 각각의 이미지에서 대응되는 특징점들을 검출, 다수의 특징점들을 통해 sub-aperture 이미지들간의 평행이동 벡터를 사전에 계산하고, 계산된 평행이동 벡터를 기반으로 이미지 정렬이 수행됐다.

3.2 이미지 융합

향상된 분해능을 얻기 위해 여러 이미지를 융합하는 방법은 여러 가지가 있지만, 본 연구에서는 실시간 처리 성능과 향상 정도를 고려해 라플라시안 피라미드 기반의 융합 알고리즘[7]을 사용했으며, 전체 과정이 Figure 4에 묘사되어 있다.

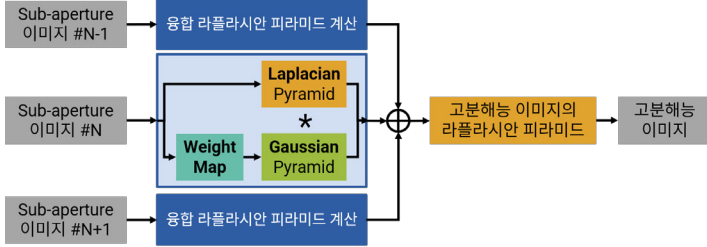


Figure 4: Overview of spatial resolution improvement algorithm.

우선 개별 sub-aperture 이미지들의 weigh map을 아래 수식을 통해 계산한다.

$$W = C^{\omega_C} \times E^{\omega_E}. \quad (2)$$

ω_C 와 ω_E 는 대비와 노출 정도에 대한 가중치이며, C 는 대비(Contrast)로, 라플라시안 필터가 적용된 이미지의 절대값을 취한 값이며, E 는 노출 정도로, 0에서 1로 표준화 된 이미지의 세기 I 인 픽셀에 대해 다음과 같이 계산된다.

$$E = \exp\left(-\frac{(I - 0.5)^2}{2\sigma^2}\right). \quad (3)$$

계산된 weight map의 가우시안 피라미드와 원본 sub-aperture 이미지의 라플라시안 피라미드[8]를 같은 레벨에 있는 이미지간에 곱셈 연산을 통해 최종적으로 융합에 필요한 sub-aperture 이미지의 라플라시안 피라미드를 얻어낼 수 있다. 위와 같은 과정을 모든 sub-aperture에 적용해 계산된 라플라시안 피라미드를 모두 합치면 분해능이 향상된 이미지의 라플라시안 피라미드가 나오게 되고, 이를 reconstruction 하면 분해능이 향상된 이미지를 얻을 수 있다.

3.3 실시간 처리

개발된 초박형 라이트필드 카메라는 임베디드 컴퓨팅 보드에서 CPU와 GPU를 모두 사용하는 구성으로 되었다. 기본적으로 이미지를 수신해 업로드 하는 부분과 다운로드해 디스플레이하는 부분은 CPU에서 처리되며, 사전에 계산된 카메라 캘리브레이션과 이미지 정렬을 적용하고 이미지 융합 알고리즘을 적용하는 부분은 GPU에서 처리된다. 이미지 연산은 C++와 GPU를 사용하는 OpenCV 라이브러리를 통해 처리되며, 연산의 대부분은 GPU를 통해 병렬적으로 처리된다. 또한 추가적으로 이미지 프레임의 처리 속도를 향상시키기 위해 OpenCV의 CUDA stream을 CPU 연산과 비동기식으로 처리하는 방식을 통해 Figure 5 과 같이 파이프라이닝 하는 방식이 사용되었다.

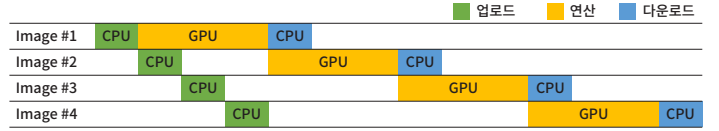


Figure 5: Asynchronized pipeline for real-time processing.

Table 1: Processing speed comparison (average fps).

단일 이미지	분해능 향상 알고리즘 (동기식)	분해능 향상 알고리즘 (비동기식)
30.00	8.17	10.65

4. 실험결과

초박형 라이트필드 카메라에서 얻어진 24장의 이미지를 중앙의 이미지(2행 2열)를 기준으로 3장의 알고리즘을 이용해 융합한 결과는 다음과 같다. Figure 6의 우측 이미지는 분해능 향상 알고리즘을 통해 촬영된 이미지들이며, 육안으로 관찰했을 때 분해능 향상 알고리즘이 적용된 이미지가 단일 이미지(Figure 6의 좌측)에 비해 품질이 우수한 것을 확인할 수 있다.

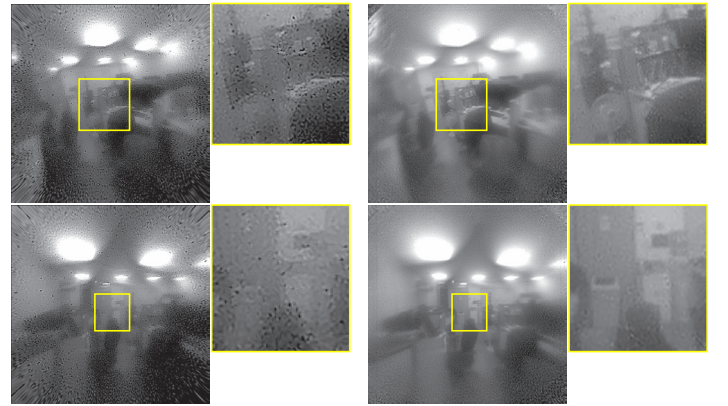


Figure 6: Qualitative result: Single image (left), Spatial resolution improved image with our algorithm (right)

Figure 7의 실험 영상을 통해서 분해능 향상 알고리즘의 정량적인 분석은 MTF50 값을 통해 수행되었다[9]. Figure 8는 ISO 12233 기준에 따라 측정된 modulation transfer function(MTF)와 MTF50을 측정한 결과이다. 측정 결과에 따르면 단일 이미지 대비 융합 이미지의 MTF50 값이 수평 방향에서 37.25%, 수직 방향에서 33.08% 개선되었음을 확인할 수 있다. Table 1는 임베디드 스템인 Nvidia Jetson NX에서 GPU를 사용해 단일 이미지를 처리하는 경우와 분해능 향상 알고리즘을 비동기식/동기식으로 처리하는 속도를 비교한 결과로, 비동기식 파이프라인의 경우 동기식 파이프라인 대비 처리속도가 30%의 증가했으며, 비동기식 파이프라인을 적용한 분해능 향상 알고리즘의 경우 10.65 fps로 실시간 처리가 가능한 것을 알 수 있다.

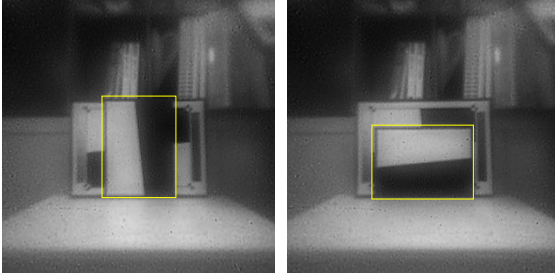


Figure 7: Image for measuring MTF: Horizontal direction (left), Vertical direction (right)

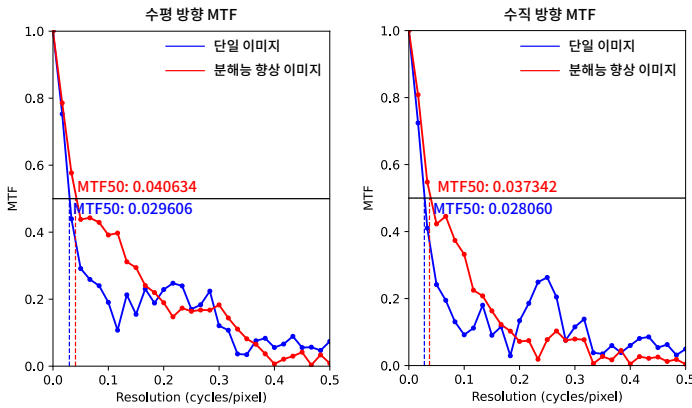


Figure 8: MTF measurement result

5. 결론

본 연구에서는 휴대가 가능한 초박형 라이트필드 카메라 시스템과 초박형 라이트필드 카메라에서 얻어지는 이미지들을 실시간으로 융합해 분해능을 향상시키는 알고리즘을 개발했다. 개발된 초박형 라이트필드 카메라는 2mm 두께이며, 임베디드 컴퓨팅 보드에서 분해능 향상 알고리즘이 10.65fps의 속도로 실시간 처리가 가능함을 확인했다. 또한 실시간 처리를 통해 획득된 이미지는 MTF50이 수평 방향에서 37.25%, 수직방향에서 33.08%로 평균 약 35% 개선이 있음을 확인했다.

추후 연구될 수 있는 부분은 다음과 같다. 현재 제작된 초박형 라이트필드 카메라의 경우 렌즈 어레이를 직접 제작했기 때문에 노이즈 발생, 빛 투과 등의 문제가 발생한다. 이 부분을 더 개선된 공정에서 제작할 경우 렌즈 어레이의 품질이 향상되고, 더 나은 품질의 이미지를 획득할 수 있을 것이다. 또한 현재 개발된 시스템의 경우 흑백 이미지만을 사용하게 되어 있으나, 추후 컬러 이미징이 가능한 카메라와 개발된 분해능 향상 알고리즘의 변경을 통해 컬러 이미지를 획득하는 시스템을 제작할 수 있을 것이다.

감사의 글

이 연구는 LIG Nex1 산학협력과제 지원으로 연구되었음.

References

- [1] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proceedings of SIGGRAPH 1996*. ACM, 1996, pp. 43–54.
- [2] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996, New Orleans, LA, USA, August 4-9, 1996*. ACM, 1996, pp. 31–42.
- [3] D. Kang, D. S. Jeon, H. Kim, H. Jang, and M. H. Kim, "View-dependent scene appearance synthesis using inverse rendering from light fields," in *Proc. IEEE International Conference on Computational Photography (ICCP) 2021*, May 2021.
- [4] K. Venkataraman, D. Lelescu, J. Duparré, A. McMahon, G. Molina, P. Chatterjee, R. Mullis, and S. K. Nayar, "PiCam: an ultra-thin high performance monolithic camera array," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 166:1–166:13, 2013.
- [5] A. Meuleman, H. Jang, D. S. Jeon, and M. H. Kim, "Real-time sphere sweeping stereo from multiview fisheye images," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.
- [6] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*. IEEE Computer Society, 2011, pp. 2564–2571.
- [7] T. Mertens, J. Kautz, and F. V. Reeth, "Exposure fusion: A simple and practical alternative to high dynamic range photography," *Comput. Graph. Forum*, vol. 28, no. 1, pp. 161–171, 2009.
- [8] J. H. Lee, I. Choi, and M. H. Kim, "Laplacian patch-based image synthesis," in *Proc. IEEE Computer Vision and Pattern Recognition (CVPR 2016)*. Las Vegas, USA: IEEE, 2016.
- [9] M. H. Kim, T. A. Harvey, D. S. Kittle, H. Rushmeier, J. Dorsey, R. O. Prum, and D. J. Brady, "3d imaging spectroscopy for measuring hyperspectral patterns on solid objects," *ACM Transactions on Graphics (Proc. SIGGRAPH 2012)*, vol. 31, no. 4, pp. 38:1–11, 2012. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=2185520.2185534>

〈 저 자 소 개 〉



김 동 진

- 2018.08: KAIST 건설 및 환경공학 학사
- 2020.08: KAIST 건설 및 환경공학 석사
- 2020.09 - 현재: KAIST 전산학부 석사과정
- 관심분야: Computer Vision, Computer Graphics, Computational Imaging
- <https://orcid.org/0000-0002-6670-6263>



유 재 관

- 2004.08: 경희대학교 기계공학과(공학석사)
- 2009.06: JAIST(일본) 로봇공학과(공학박사)
- 2009.07 - 현재: LIG넥스원 미래기술연구소 팀장
- 관심분야: Unmanned & Robotic systems, Insect robot, Behavior analysis
- <https://orcid.org/0000-0003-4176-3867>



조 용 진

- 2013.02 한양대학교 전자시스템공학과(공학석사)
- 2013 - 현재: LIG넥스원 미래기술연구소 선임연구원
- 관심분야: AI, Image Enhancement, Object Detection & Tracking
- <https://orcid.org/0000-0002-3749-7570>



김 민 혁

- 2010.02: University College London (UCL) 전산학 박사
- 2010.08 - 2012. 08: Yale University 박사 후 연구원
- 2012.08 - 현재: KAIST 전산학부 교수
- 2019.03 - 현재: KAIST 전산학부 석좌교수
- 관심분야: Computational Imaging & Computational Photography, Image-Based Modeling and Rendering, Color & Visual Perception
- <https://orcid.org/0000-0002-5078-4005>