

사진측량과 컴퓨터비전 간의 카메라 렌즈왜곡 변환 Conversion of Camera Lens Distortions between Photogrammetry and Computer Vision

홍송표¹⁾ · 최한승²⁾ · 김의명³⁾

Hong, Song Pyo · Choi, Han Seung · Kim, Eui Myoung

Abstract

Photogrammetry and computer vision are identical in determining the three-dimensional coordinates of images taken with a camera, but the two fields are not directly compatible with each other due to differences in camera lens distortion modeling methods and camera coordinate systems. In general, data processing of drone images is performed by bundle block adjustments using computer vision-based software, and then the plotting of the image is performed by photogrammetry-based software for mapping. In this case, we are faced with the problem of converting the model of camera lens distortions into the formula used in photogrammetry. Therefore, this study described the differences between the coordinate systems and lens distortion models used in photogrammetry and computer vision, and proposed a methodology for converting them. In order to verify the conversion formula of the camera lens distortion models, first, lens distortions were added to the virtual coordinates without lens distortions by using the computer vision-based lens distortion models. Then, the distortion coefficients were determined using photogrammetry-based lens distortion models, and the lens distortions were removed from the photo coordinates and compared with the virtual coordinates without the original distortions. The results showed that the root mean square distance was good within 0.5 pixels. In addition, epipolar images were generated to determine the accuracy by applying lens distortion coefficients for photogrammetry. The calculated root mean square error of y-parallax was found to be within 0.3 pixels.

Keywords : Photogrammetry, Computer Vision, Interior Orientation Parameters, Lens Distortions, Epipolar Images

초 록

사진측량과 컴퓨터비전 분야는 카메라에서 촬영된 영상에서 3차원 좌표를 결정하는 것은 동일하지만 두 분야는 카메라 렌즈왜곡 모델링 방법과 카메라 좌표계의 차이점으로 인하여 서로 간에 직접적인 호환이 어렵다. 일반적으로 드론 영상의 자료처리는 컴퓨터비전 기반의 소프트웨어를 이용하여 번들블록조정을 수행한 후 지도제작을 위해서 사진측량 기반의 소프트웨어로 도화를 수행하게 된다. 이때 카메라 렌즈왜곡의 모델을 사진측량에서 사용하는 수식으로 변환해야 하는 문제에 직면하게 된다. 이에 본 연구에서는 사진측량과 컴퓨터비전에서 사용되는 좌표계와 렌즈왜곡 모델식의 차이점에 대하여 기술하고 이를 변환하는 방법론을 제안하였다. 카메라 렌즈왜곡 모델의 변환식의 검증을 위해서 먼저 렌즈왜곡이 없는 가상의 좌표에 컴퓨터비전 기반의 렌즈왜곡 모델을 이용하여 렌즈왜곡을 부여하였다. 그리고 나서 렌즈왜곡이 부여된 사진좌표를 이용하여 사진측량 기반의 렌즈왜곡 모델을 이용하여 왜곡계수를 결정한 후 사진좌표에서 렌즈왜곡을 제거하여 원래의 왜곡이 없는 가상좌표와 비교하였다. 그 결과 평균제곱근거리가 0.5픽셀 이내로 양호한 것으로 나타났다. 또한 사진측량용 렌즈왜곡 계수를 적용하여 정밀도화 가능여부를 판단하기 위해서 에피폴라 영상을 생성하였다. 생성된 에피폴라 영상에서 y-시차의 평균제곱근오차가 계산한 결과 0.3픽셀 이내로 양호하게 나타났음을 알 수 있었다.

핵심어 : 사진측량, 컴퓨터비전, 내부표정요소, 렌즈왜곡, 에피폴라 영상

Received 2019. 08. 19, Revised 2019. 08. 27, Accepted 2019. 08. 28

1) Dept. of GIS Engineering, Namseoul University (E-mail: bbcuco@gmail.com)

2) Member, Mapping & Localization, Naver Labs Corp. (E-mail: choi.hs@naverlabs.com)

3) Corresponding Author, Member, Department of Spatial Information Engineering, Namseoul University (E-mail: kemyoung@nsu.ac.kr)

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

최근 컴퓨터비전 기술의 발달로 인하여 정사영상, 3차원 점군(point cloud), 3차원 모델 등의 결과물을 자동화된 방법으로 제작할 수 있기 때문에 사용자 측면에서는 공간정보를 보다 손쉽게 취득할 수 있게 되었다. 그 이유는 전통적인 항공사진측량 기술을 적용한 소프트웨어는 알고리즘의 매개변수 설정, 도화작업, 품질평가 등의 전문적인 지식을 요구하는 반면에 컴퓨터비전 기술을 이용한 소프트웨어는 사용자의 편의성을 고려하여 자동화와 신속성에 초점을 맞추고 있기 때문이다(Verhoeven *et al.*, 2012; Rabiou and Waziri, 2014).

컴퓨터비전 기술을 이용한 공간정보 제작에 관한 연구사례로는 고해상도의 DMC (Digital Mapping Camera)에서 취득한 대용량의 항공영상을 컴퓨터비전 기반의 소프트웨어를 이용하여 실감정사영상(true ortho photo)의 제작 가능성을 분석한 연구(Kim *et al.*, 2017), 드론 영상과 컴퓨터비전 기반의 번들블록조정법(BBA: Bundle Block Adjustment)으로 결정된 내·외부표정요소를 이용하여 대축척 수치지도의 제작 가능성을 분석한 연구(Kim *et al.*, 2019; Lee *et al.*, 2019) 등이 수행되었으며 산출된 대부분의 공간정보가 국내 법령에 따른 허용오차 이내인 것으로 분석되었다(Lim *et al.*, 2015).

컴퓨터비전 기술을 이용하여 영상에서 공간정보를 제작할 수 있는 대표적인 상용 소프트웨어로는 Pix4D Mapper, Photoscan, Context Capture 등이 있고 오픈 소프트웨어 및 오픈 소스로는 COLMAP, Open Drone Map, Visual SFM, Theia, OpenMVG, OpenCV 등이 있으며 사진측량 분야에서도 광범위하게 활용되고 있다(Hong, 2016; Bianco *et al.*, 2018). 하지만 컴퓨터 비전과 사진측량에서 사용하고 있는 내부표정요소에 해당하는 렌즈왜곡 모델식과 좌표체계에서 서로 다른 차이점이 있다(Mikhail *et al.*, 2001; Hartley and Zisserman 2003; Luhmann *et al.*, 2011; McGlone, 2013; Drap and Lefèvre, 2016; Kaehle and Bradski, 2016).

컴퓨터비전 기술을 이용하여 내·외부표정요소를 결정한 결과를 이용하여 지도제작을 위한 작업을 수행하기 위해서는 도화기능이 소프트웨어에 탑재되어 있어야 한다. 그러나 대부분의 컴퓨터비전 기반의 소프트웨어는 도화기능이 탑재되어 있지 않아 수치지도를 제작하기 위해서는 이러한 기능을 지원하는 SOCET SET, UASMaster 등과 같은 사진측량 기반의 소프트웨어를 사용하여야 한다. 이때 컴퓨터비전 기술의 소프트웨어로 처리한 카메라의 렌즈왜곡 모델을 사진측량 기반의 소프트웨어에 적용하기 위해서는 렌즈왜곡 모델의 매개변수와 데이터 처리방식의 차이로 인해서 이를 변환해야 한다.

이에 본 연구에서는 컴퓨터비전 기술을 적용하여 결정된 카메라의 렌즈왜곡 모델을 사진측량에서 사용하고 있는 렌즈왜곡 모델로 변환할 수 있는 방법을 제안하는 것으로 이를 통해 도화작업을 원활하게 수행할 수 있도록 지원하는 것이 목적이다.

2. 연구방법

2.1 사진측량과 컴퓨터비전의 좌표계

사진측량에서는 Fig. 1과 같이 픽셀좌표계(pixel coordinate system)와 사진좌표계(photo coordinate system)가 존재하며, 수치적 안정성(numerical stability)을 위해 Eq. (1)과 같이 픽셀좌표를 사진좌표로 변환한 후, 기하학적 연산을 수행한다. 또한 픽셀좌표의 경우에는 픽셀단위를 사용하며, 사진좌표는 일반적으로 mm단위를 사용하지만 픽셀의 크기를 $1(mm/pix)$ 로 설정할 경우에는 픽셀단위가 된다. 또한 주점(x_p, y_p) 및 초점거리는 사진좌표계에서 정의된 값을 사용한다(Mikhail *et al.*, 2001; Luhmann *et al.*, 2011; McGlone, 2013).

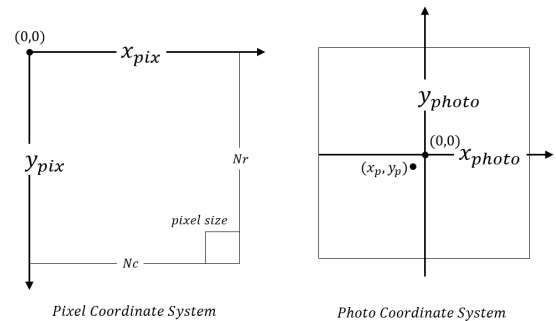


Fig. 1. Coordinate system in photogrammetry

$$\begin{aligned} x_{photo} &= (x_{pix} - \frac{N_c}{2}) x_{pix\ size} \\ y_{photo} &= -(y_{pix} - \frac{N_r}{2}) y_{pix\ size} \end{aligned} \quad (1)$$

where, N_c is the number of pixels in a column, N_r is the number of pixels in a row, $x_{pix\ size}$ is the pixel size of x , $y_{pix\ size}$ is the pixel size of y , (x_{photo}, y_{photo}) is a photo coordinate, (x_{pix}, y_{pix}) is a pixel coordinate.

컴퓨터비전에서는 Fig. 2와 같이 픽셀좌표계와 정규좌표계(normalized pixel coordinate system)가 존재하며, 컴퓨터비전에서의 픽셀좌표계는 사진측량에서 사용하는 픽셀좌표계와 동일하지만, 주점 및 초점거리의 경우에는 픽셀좌표계에

서 정의된다.

또한 수치적 안정성을 위해 Eq. (2)와 같이 픽셀좌표를 정규좌표로 변환한 후 기하학적 연산을 수행한다(Hartley and Zisserman, 2003; Kaehler and Bradski, 2016). 정규좌표는 픽셀좌표를 초점거리로 나누었기 때문에 단위가 없는 무차원 상수값이 된다. 컴퓨터 비전에서 정규좌표를 사용하는 이유는 연산과정에서의 효율성을 위하여 동차좌표(homogeneous coordinate)를 사용하기 때문에 초점거리를 1로 정규화 하면서 수치적 안정성이 보장되는 좌표계에서 연산처리를 하는 것이 수월하기 때문이다(Hartley and Zisserman 2003).

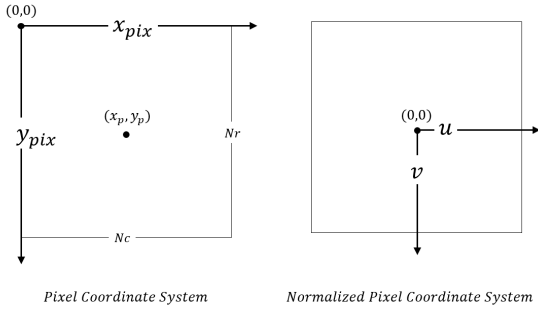


Fig. 2. Coordinate system in computer vision

$$\begin{aligned} u &= \frac{(x_{pix} - x_p)}{f} \\ v &= \frac{(y_{pix} - y_p)}{f} \end{aligned} \quad (2)$$

where, (x_{pix}, y_{pix}) is a pixel coordinate, f is a focal length, (x_p, y_p) is a principal point in pixel coordinate system, and (u, v) is a normalized pixel coordinate.

2.2 사진측량과 컴퓨터비전의 렌즈왜곡 차이점

사진측량의 렌즈왜곡 모델식은 Eq. (3)과 같이 렌즈왜곡이 부여된 사진좌표(x_{add}, y_{add})에 렌즈왜곡 계수를 이용하여 왜곡이 없는 사진좌표(x_{free}, y_{free})로 변환한다(Mikhail *et al.*, 2001; Luhmann *et al.*, 2011; McGlone, 2013).

$$\begin{aligned} \begin{bmatrix} x_{free} \\ y_{free} \end{bmatrix} &= (1 - k_1 r^2 - k_2 r^4 - k_3 r^6) \begin{bmatrix} \bar{x}_{add} \\ \bar{y}_{add} \end{bmatrix} \\ &\quad - \begin{bmatrix} p_1 (r^2 + 2\bar{x}_{add}^2) + 2p_2 \bar{x}_{add} \bar{y}_{add} \\ 2p_1 \bar{x}_{add} \bar{y}_{add} + p_2 (r^2 + 2\bar{y}_{add}^2) \end{bmatrix} \end{aligned} \quad (3)$$

where, (x_{free}, y_{free}) is an undistorted point in photo coordinate system, (x_{add}, y_{add}) is a distorted point in photo coordinate system, (x_p, y_p) is a principal point in photo coordinate system, $x_{add} = x_{pix} - x_p$, $y_{add} = y_{pix} - y_p$, $r = \sqrt{x_{add}^2 + y_{add}^2}$ and radial distance, (k_1, k_2, k_3) are radial distortion parameters in photo coordinate system, and (p_1, p_2) are decentric distortion parameters in photo coordinate system.

하지만 컴퓨터비전에서 사용하는 렌즈왜곡 모델식은 사진측량과 달리 렌즈왜곡이 없는 픽셀좌표를 정규좌표(u_{free}, v_{free})로 변환한 후 Eq. (4)와 같이 렌즈왜곡 계수를 이용하여 렌즈왜곡을 고려한 정규좌표(u_{add}, v_{add})로 변환한다(Hartley and Zisserman, 2003; Kaehler and Bradski, 2016).

$$\begin{aligned} \begin{bmatrix} u_{add} \\ v_{add} \end{bmatrix} &= (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \begin{bmatrix} u_{free} \\ v_{free} \end{bmatrix} \\ &\quad + \begin{bmatrix} 2p_1 u_{free} v_{free} + p_2 (r^2 + 2u_{free}^2) \\ p_1 (r^2 + 2v_{free}^2) + 2p_2 u_{free} v_{free} \end{bmatrix} \end{aligned} \quad (4)$$

where, u_{free}, v_{free} is an undistorted point in normalized pixel coordinate system, u_{add}, v_{add} is a distorted point in normalized pixel coordinate system, $r = \sqrt{u_{free}^2 + v_{free}^2}$ and radial distance, (k_1, k_2, k_3) are radial distortion parameters in normalized pixel coordinate system, and (p_1, p_2) are decentric distortion parameters in normalized pixel coordinate system.

이처럼 사진측량과 컴퓨터비전의 렌즈왜곡모델 기법이 서로 다른 이유는 사진측량은 컴퓨터를 이용한 영상처리가 발전하기 이전부터 연구되었던 학문으로 관측자가 렌즈왜곡이 있는 원 영상으로 부터 수동으로 공액점(conjugate point)을 관측하여 렌즈왜곡을 제거한 후, 대상물의 3차원 위치를 결정하는 것이 목적이었다. 따라서 Eq. (3)과 같이 카메라의 렌즈왜곡 계수를 알고 있다면 렌즈왜곡을 제거하는 과정을 선행식으로 정의할 필요가 있었던 것이다.

반면 컴퓨터비전에서는 연산처리의 자동화 및 속도에 중점을 두었기 때문에 왜곡이 있는 원 영상에서 렌즈왜곡이 제거된 영상을 제작한 후, 이를 특징점 매칭을 수행하여 가상의 3차원 위치를 신속하게 결정하는 것이 목적이었다. 이를 위해 Eq. (4)와 같은 선형식을 이용하여 렌즈왜곡이 없는 좌표에 렌

즈왜곡을 부여하고 렌즈왜곡이 있는 원 영상에서 주변의 밝기 값을 가져와서 보간하는 방법을 사용한다.

사진측량의 렌즈왜곡 모델은 렌즈왜곡이 없는 사진좌표에서 렌즈왜곡을 부여할 경우는 비선형식이 되므로 수치적 반복법을 통하여 렌즈왜곡을 부여하는 방법을 사용하지만, 컴퓨터비전에서는 렌즈왜곡을 부여하는 과정을 선형식으로 정의하여야만 연산속도 측면에서 효율적이기 때문이다. 뿐만 아니라 사진측량에서의 렌즈왜곡 계수는 사진좌표계에서 정의되고 컴퓨터비전에서의 렌즈왜곡계수는 정규좌표계에서 정의된다. 또한 편심왜곡(decentric distortion) 계수의 경우 p_1 , p_2 가 p_2 , p_1 으로 계수가 반대로 되어 있다는 것에 주의해야 한다. 그렇기 때문에 데이터처리 소프트웨어 및 오픈소스에서 정의하고 있는 렌즈왜곡 모델식 및 좌표계를 정확하게 인지하고 사용하여야 한다.

2.3 컴퓨터비전에서 사진측량으로의 렌즈왜곡 계수 변환

Fig. 3은 컴퓨터비전에서 사진측량으로의 렌즈왜곡 계수를 산출하기 위한 과정을 나타낸 것이다.

- 1) 컴퓨터비전 소프트웨어를 이용하여 결정한 카메라 렌즈 왜곡 계수는 왜곡이 없는 좌표에서 왜곡을 부여하는 좌표로 계산한 것이기 때문에, 영상의 크기를 고려하여 렌즈왜곡이 없다고 가정한 가상의 격자점을 생성한다.
- 2) 영상에 생성한 가상의 격자점은 렌즈왜곡이 없는 좌표이므로 컴퓨터비전 수식을 이용하여 결정한 렌즈왜곡 계수를 이용하여 왜곡을 부여하여 사진좌표로 변환한다.
- 3) 렌즈왜곡이 없는 사진좌표(가상의 격자점)와 렌즈왜곡이 부여된 사진좌표를 입력값으로 하여 사진측량에서 사용하는 렌즈왜곡 계수를 선형식을 이용하여 구한다. 다시 말하면 렌즈왜곡이 부여된 사진좌표에서 렌즈왜곡이 없는 사진좌표로의 렌즈왜곡 계수를 결정한다.

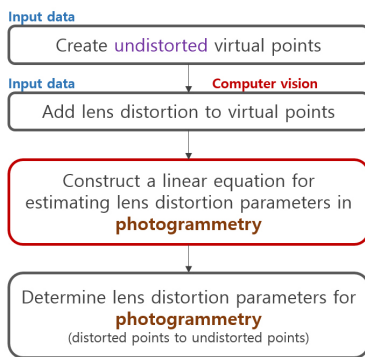


Fig. 3. Flow chart for conversion to lens distortion parameters for photogrammetry

컴퓨터비전에서 결정한 렌즈왜곡 계수를 사진측량에서 사용하는 렌즈왜곡 계수로 변환하기 위하여 Eq. (3)을 Eq. (5)와 같이 사진측량의 렌즈왜곡 계수를 미지수로 하는 선형방정식을 구성한다. 여기서 렌즈왜곡이 없는 사진좌표(x_{free}, y_{free})와 주점의 위치를 고려한 왜곡이 있는 사진좌표(x_{add}, y_{add})는 영상 크기 내에서 왜곡 없다고 가정한 가상의 격자점(정규좌표)들을 생성하고, Eq. (4)의 컴퓨터비전 렌즈왜곡모델 식을 이용하여 왜곡을 부여하고 이를 사진좌표로 변환한다.

$$\begin{bmatrix} r^2 \bar{x}_{add} & r^4 \bar{x}_{add} & r^6 \bar{x}_{add} & (r^2 + 2\bar{x}_{add}^2) & 2\bar{x}_{add}\bar{y}_{add} \\ r^2 \bar{y}_{add} & r^4 \bar{y}_{add} & r^6 \bar{y}_{add} & 2\bar{x}_{add}\bar{y}_{add} & (r^2 + 2\bar{y}_{add}^2) \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} \bar{x}_{add} - x_{free} \\ \bar{y}_{add} - y_{free} \end{bmatrix} \quad (5)$$

Eq. (5)를 행렬 기호로 표현하면 Eq. (6)과 같고 Eq. (7)을 이용하여 사진측량에서 사용하는 렌즈왜곡 계수로 변환한다. 이때 한 점당 2개의 관측방정식이 구성되고, 미지수는 렌즈왜곡 계수를 고려할 때 5개 이므로 최소 3점 이상이 필요하다.

$$Ax = b \quad (6)$$

$$x = N^{-1}b \quad (7)$$

$$\text{where, } N = A^T A, b = A^T b, x = [k_1 \ k_2 \ k_3 \ p_1 \ p_2]^T$$

이때 Eq. (5)를 살펴보면 영상의 크기가 크고, 단위가 픽셀인 사진좌표를 사용할 경우에는 방사거리(r)의 영향으로 Eq. (7)의 행렬 N 의 값이 매우 커져, 역행렬을 계산할 수 없는 경우가 발생할 수 있다. 따라서 Eq. (8)과 같이 영상내의 최대 방사거리(r_{\max})를 각 사진좌표에 나누어 무단위의 사진좌표로 정규화 시킨다.

$$r_{\max} = \sqrt{\left(\frac{H}{2}\right)^2 + \left(\frac{W}{2}\right)^2} \quad (8)$$

where, H is the height of an image in photo coordinate system, W is the width of an image in photo coordinate system.

최종적으로 Eq. (9)와 같이 무단위의 정규화된 사진좌표에서 추정된 렌즈왜곡 계수를 정규화 하였던 최대 방사거리를 이용하여 단위를 복원시킨다.

$$\begin{aligned}
k_1 &= \hat{p}_1 r_{\max}^2 \\
k_2 &= \hat{p}_2 r_{\max}^4 \\
k_3 &= \hat{p}_3 r_{\max}^6 \\
p_1 &= \hat{p}_1 r_{\max}^2 \\
p_2 &= \hat{p}_2 r_{\max}^2
\end{aligned} \tag{9}$$

where, $(\hat{k}_1, \hat{k}_2, \hat{k}_3, \hat{p}_1, \hat{p}_2)$ are distortion parameters in normalized photo coordinate system.

2.4 사진측량에서 컴퓨터비전으로의 렌즈왜곡 계수 변환

Fig. 4는 사진측량에서 컴퓨터비전으로의 렌즈왜곡 계수를 산출하기 위한 과정을 나타낸 것이다.

- 1) 사진측량 소프트웨어를 이용하여 결정한 카메라 렌즈왜곡 계수는 왜곡이 있는 사진좌표를 렌즈왜곡을 제거한 사진좌표로 변환할 수 있는 계수이므로 먼저 렌즈왜곡이 있다고 가정된 가상의 격자점을 영상의 크기를 고려하여 생성한다.
- 2) 영상에 생성한 렌즈왜곡이 있는 가상의 격자점에서 사진측량 수식을 이용하여 결정한 렌즈왜곡 계수를 이용하여 렌즈왜곡이 제거된 영상좌표로 변환한 후 초점거리를 이용하여 정규좌표로 변환한다.
- 3) 렌즈왜곡이 있는 정규좌표와 렌즈왜곡을 제거한 정규좌표를 입력값으로 하여 컴퓨터비전에서 사용하는 렌즈왜곡 계수를 선형식을 이용하여 구한다. 다시 말하면 렌즈왜곡이 없는 정규좌표에서 렌즈왜곡이 부여된 정규좌표로의 렌즈왜곡 계수를 결정한다.

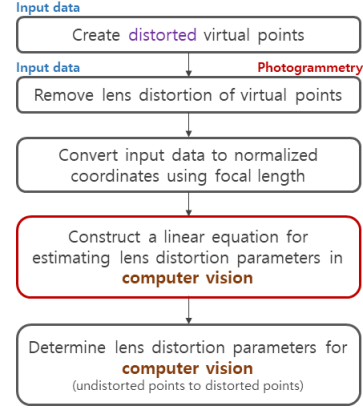


Fig. 4. Flow chart for conversion to lens distortion parameters for computer vision

사진측량의 렌즈왜곡 계수를 컴퓨터비전에 사용하는 렌즈왜곡 계수로 변환하기 위하여 Eq. (4)를 Eq. (10)과 같이 컴퓨터비전의 렌즈왜곡 계수를 미지수로 하는 선형방정식으로 구성한다. 여기서 왜곡이 없는 정규좌표(u_{free}, v_{free})와 왜곡이 있는 정규좌표(u_{add}, v_{add})는 2.3절과 달리 영상 크기 내에서 왜곡이 있다고 가정된 가상의 격자점(사진좌표)들을 생성하고 Eq. (3)의 사진측량 렌즈왜곡모델 식을 이용하여 렌즈왜곡을 제거하고 이를 정규좌표로 변환한다.

$$\begin{bmatrix} r^2 u_{free} & r^4 u_{free} & r^6 u_{free} & 2u_{free}v_{free} & (r^2 + 2u_{free}^2) \\ r^2 v_{add} & r^4 v_{free} & r^6 v_{free} & (r^2 + 2v_{free}^2) & 2u_{free}v_{free} \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} u_{add} - u_{free} \\ v_{add} - v_{free} \end{bmatrix} \tag{10}$$

Eq. (10)을 이용하여 결정한 컴퓨터비전에서 사용하는 렌즈왜곡 계수는 데이터 처리과정에서 이미 정규좌표를 사용하여 엮기 때문에 Eq. (7)을 이용할 때와 달리 렌즈왜곡 계수의 단위를 복원할 필요가 없다.

3. 실험

3.1. 컴퓨터비전용 소프트웨어를 이용한 내부표정요소 결정 및 변환

카메라 렌즈왜곡 계수에 대한 실험을 위하여 충남 천안시 서북면 일대를 DJI사의 회선익 드론인 Matrice 100에 장착된 Zenmuse X3 카메라를 이용하여 702장의 영상을 취득하였고 촬영된 면적은 약 707m×1050m이다.

촬영된 702장의 드론영상과 가상기준점(VRS: Virtual Reference Station)측량 방식으로 취득한 9점의 지상기준점을 Pix4D Mapper의 BBA를 이용하여 카메라의 내·외부표정요소를 결정하였고 내부표정요소는 Table 1과 같다. 이때 Pix4D Mapper는 컴퓨터비전용 소프트웨어 이므로 BBA로 결정된 내부표정요소는 Eq. (4)와 같은 컴퓨터비전 수식을 따른다(Pix4Dmapper, 2019).

Table 1. IOP for computer vision determined using Pix4D Mapper

	Element	Value	Unit
Focal length	f	8362.907	<i>pixel</i>
Principal point	x_p	2033.970	<i>pixel</i>
	y_p	1476.135	<i>pixel</i>
Image size	<i>width</i>	4000	<i>pixel</i>
	<i>height</i>	3000	<i>pixel</i>
Radial distortion	k_1	8.660652e-02	-
	k_2	-1.414601e+00	-
	k_3	8.242845e+00	-
Decentric distortion	p_1	-1.816357e-04	-
	p_2	7.853989e-04	-

BBA의 정확도 검증은 가상기준점 측량 방식으로 취득한 8개의 검사점과 결정된 내·외부표정요소를 이용하여 평균 재투영오차(mean re-projection error)를 계산하였으며 약 0.405 픽셀로 양호하게 나타났다.

또한 렌즈왜곡의 정도를 파악하기 위하여 Fig. 5와 같이 왜곡이 없다고 가정한 841개의 가상의 격자점을 생성하고 Table의 컴퓨터비전의 내부표정요소를 Eq. (4)를 이용하여 왜곡을 부여하였다. 이때 렌즈왜곡의 영향만을 파악위하여 주점을 고려하지 않았다. 각 좌표의 렌즈왜곡량을 정량적으로 파악하기 위하여 Eq. (11), (12)를 이용하여 Table 2와 같이 평균제곱근오차(RMSE: Root Mean Square Error)와 평균제곱근거리(RMSD: Root Mean Square Distance)를 계산하였으며, x와 y방향의 왜곡량이 1~2 픽셀이내인 것을 알 수 있다. 이를 통해 드론에 장착된 Zenmuse X3 카메라의 렌즈왜곡은 비교적 크지 않은 것을 알 수 있었다.

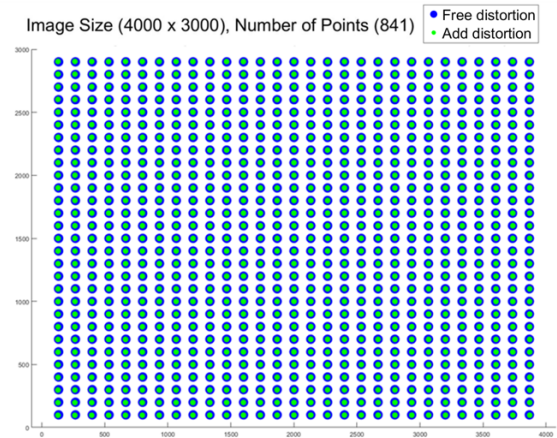


Fig. 5. Effect of lens distortion factors on Zenmuse X3 camera

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (p_i - a_i)^2}{n}} \quad (11)$$

where, n is the number of points, p is a predicted value, a is an actual value.

$$RMSD = \sqrt{\frac{\sum_{i=1}^n (p_i - a_i)^T (p_i - a_i)}{n}} \quad (12)$$

where, n is the number of points, p is a predicted vector, a is an actual vector.

Table 2. Analysis of pixel coordinates with distortion (unit: pixel)

RMSE(x)	RMSE(y)	RMSD(x,y)
1.943227	1.323583	2.351171

Table 1의 컴퓨터비전 렌즈왜곡 계수를 2.3절에 기술된 방법을 이용하여 Table 3과 같이 사진측량에서 사용하는 렌즈 왜곡 계수로 변환하였고, 사후분산($\hat{\sigma}_0^2$)은 Eq. (13)과 같이 계산하였다. 일관적인 해석을 위하여 사진좌표의 단위는 픽셀단위를 사용하였다.

$$\hat{\sigma}_0^2 = \frac{e^T e}{2n - r} \quad (13)$$

where, n is the number of points, r is the number of unknown parameters, e is an error vector($2n \times 1$).

Table 3. Lens distortion parameters for photogrammetry

	Element	Value	Unit
Posterior variance	$\hat{\sigma}_0^2$	2.476177e-07	$pixel^2$
Focal length	f	8362.907	$pixel$
Principal point	x_p	33.970	$pixel$
	y_p	23.865	$pixel$
Radial distortion	k_1	1.233875e-09	$pixel^{-2}$
	k_2	-2.877473e-16	$pixel^{-4}$
	k_3	2.392324e-23	$pixel^{-6}$
Decentric distortion	p_1	-3.733204e-11	$pixel^{-2}$
	p_2	8.635360e-12	$pixel^{-2}$

렌즈왜곡 계수를 변환한 결과의 검증을 위하여 Fig. 6과 같이 왜곡 없다고 가정하 가상의 격자점(검사점)을 생성하였으며, Table 1의 컴퓨터비전용 내부표정요소를 Eq. (4)에 적용하여 왜곡을 부여하였다. 그 후 왜곡이 부여된 점에 Table 3에서 결정된 사진측량에서의 카메라 렌즈왜곡 계수를 Eq. (3)과 같이 적용하여 왜곡을 제거하였다.

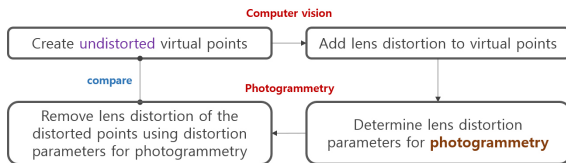
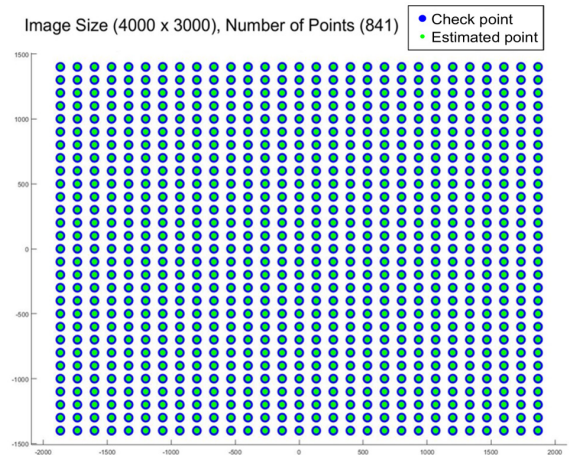
**Fig. 6. Procedures for algorithm verification**

Fig. 7은 처음 생성한 가상의 격자점인 검사점과 사진측량용 렌즈왜곡 계수를 이용하여 왜곡의 제거하여 추정된 점을 좌표평면에 출력한 것이며, 렌즈왜곡의 영향만을 파악하기 위하여 주점은 고려하지 않았다.

Table 4는 Eq. (12)와 Eq. (13)을 이용하여 RMSE와 RMSD를 계산한 결과, RMSE는 약 0.4픽셀이내로 나타났고, RMSD는 약 0.432픽셀로 나타났다.

**Fig. 7. Location of checkpoints and estimated points****Table 4. Error analysis of transformed coordinate points (unit: pixel)**

RMSE(x)	RMSE(y)	RMSD(x,y)
0.212339	0.376105	0.431906

3.2 카메라 캘리브레이션 툴박스를 이용한 내부표정 요소 결정 및 변환

추가적인 실험을 위하여 Matlab 소프트웨어에서 지원하는 컴퓨터비전 기반의 카메라 캘리브레이션 툴박스를 이용하였다. 이를 위해 칼텍(caltech: california institute of technology)의 컴퓨터비전 연구실에서 제공하는 18장의 체스보드 영상을 사용하였고, 체스보드는 Fig. 8과 같이 흑백 간격은 가로 및 세로가 3cm로 구성되어 있다(Bouguet, 2015; Hong *et al.*, 2019).

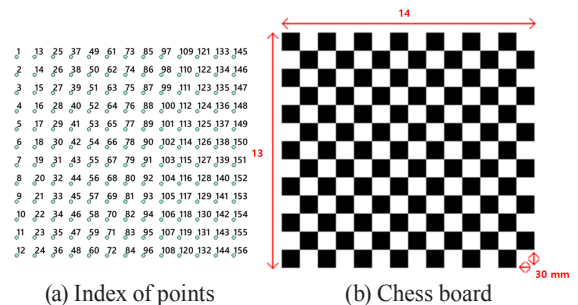
**Fig. 8. Calibration target**

Table 5는 카메라 캘리브레이션 툴박스를 이용하여 카메라 캘리브레이션을 수행한 결과값을 나타낸다. 카메라 캘리브레이션의 정확도는 Fig. (a)와 같이 156개의 검사점과 캘리브레이션을 통해 결정된 내·외부표정요소를 이용하여 평균 재투영 오차를 계산하였으며 그 결과 약 0.160 픽셀로 양호하게 나타났다.

Table 5. Camera calibration result in computer vision using Matlab

	Element	Value	Unit
Focal length	f	657.6682	<i>pixel</i>
Principal point	x_p	304.1098	<i>pixel</i>
	y_p	244.8333	<i>pixel</i>
Image size	<i>width</i>	640	<i>pixel</i>
	<i>height</i>	480	<i>pixel</i>
Radial distortion	k_1	-0.2458	-
	k_2	0.0555	-
	k_3	0.1612	-
Decentric distortion	p_1	3.6736e-06	-
	p_2	1.6723e-04	-

3.1절과 유사하게 100개의 가상 격자점을 이용하여 Fig. 9와 같이 렌즈왜곡 모델식을 이용하여 왜곡을 부여하고 그 차이를 Table 6과 같이 계산하였다. Fig. 9와 Table 6에서 알 수 있듯이 칼텍에서 제공하는 카메라 영상의 왜곡의 RMSD는 약 10픽셀 정도로 왜곡이 심한 것을 알 수 있었다.

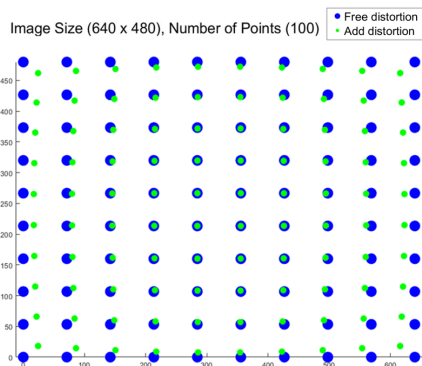


Fig. 9. Effect of lens distortion

Table 6. Analysis of pixel coordinates with distortion (unit: pixel)

RMSE(x)	RMSE(y)	RMSD(x,y)
10.701330	7.155255	12.873078

렌즈왜곡 계수를 변환한 결과의 검증은 Fig. 6에 기술한 방법과 동일하며, 3.1절과 유사하게 왜곡 없다고 가정한 가상의 격자점(검사점)에 대하여 Table 5를 이용하여 왜곡을 부여한 후 사진측량용으로 변환된 Table 7의 렌즈왜곡 계수를 이용하여 왜곡을 제거하였다.

처음 생성한 가상의 격자점인 검사점과 사진측량용 렌즈왜곡 계수를 이용하여 왜곡의 제거하여 추정된 점에 대한 RMSD는 0.045픽셀로 매우 양호한 결과가 나왔다. 이러한 이유는 실내에서 카메라 캘리브레이션을 위한 최적의 조건으로 영상을 촬영하였고, 실외에서 촬영한 드론영상의 경우, 9개의 기준점을 사용하였지만 실내에서 촬영한 영상의 경우에는 156개의 기준점을 사용하였기 때문이다.

Table 7. Lens distortion parameters for photogrammetry

	Element	Value	Unit
Posterior variance	$\hat{\sigma}_0^2$	4.568057e-05	<i>pixel</i> ²
Focal length	f	657.6682	<i>pixel</i>
Principal point	x_p	-15.8902	<i>pixel</i>
	y_p	-4.8333	<i>pixel</i>
Radial distortion	k_1	-5.528005e-07	<i>pixel</i> ⁻²
	k_2	-1.234020e-12	<i>pixel</i> ⁻⁴
	k_3	6.797313e-18	<i>pixel</i> ⁻⁶
Decentric distortion	p_1	8.302851e-10	<i>pixel</i> ⁻²
	p_2	-1.770692e-11	<i>pixel</i> ⁻²

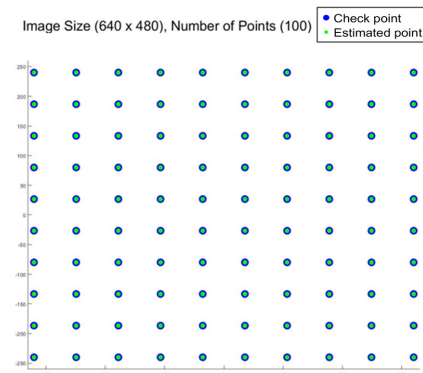


Fig. 10. Location of checkpoints and estimated points

Table 8. Error analysis of transformed coordinate points (unit: pixel)

RMSE(x)	RMSE(y)	RMSD(x,y)
0.022426	0.039034	0.045018

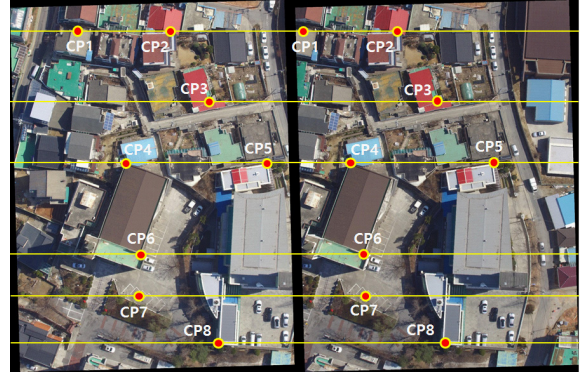
3.3 에피폴라 영상 제작 및 분석

Table 3에 나타난 것과 같이 컴퓨터비전의 내부표정요소를 사진측량에서 사용하는 내부표정요소로 변환한 결과와 Pix4D Mapper를 이용하여 결정된 외부표정요소를 이용하여 사진측량 기반의 정밀도화가 가능한지 검증하기 위해 Fig. 11의 좌우측 영상을 사진측량 기반의 수식을 이용하여 에피폴라 영상을 제작하였으며 그 결과는 Fig. 12와 같다.

**Fig. 11. Original images**

에피폴라 영상은 방향코사인을 이용한 방법을 사용하였으며, 에피폴라 영상을 제작하기 위해서는 카메라의 외부표정요소를 이용하여 y-시차를 제거할 수 있는 에피폴라 회전행렬을 계산할 뿐만 아니라 영상을 재배열(resampling)하기 위해서 렌즈왜곡 계수를 포함한 카메라의 내부표정요소도 필요하다(Kim *et al.*, 2018). 이러한 이유로 제작된 에피폴라 영상에서 y-시차가 발생하지 않는다면, 컴퓨터비전 소프트웨어에서 결정된 렌즈왜곡 계수를 본 연구에서 제안한 방법을 통해서 사진측량용 렌즈왜곡 계수로 적합하게 변환된 것을 다시 한번 확인할 수 있다.

이를 위해 Fig. 12에서 영상의 가운데 부분 뿐만 아니라 방사방향 왜곡에 많은 영향을 받는 영상의 모서리 부분에서도 검사점을 추출하였으며 8개에 대해 좌우 에피폴라 영상에서 수작업으로 y-시차를 계산하였으며 RMSE는 0.286 픽셀로 매우 양호한 결과를 보였다.

**Fig. 12 . Epipolar images****Table 9. y-parallax in epipolar images (unit: pixel)**

ID	y-left	y-right	y-parallax
CP1	359.6265	359.0843	0.54217
CP2	340.2771	340.7791	-0.50201
CP3	1119.904	1120.06	-0.15663
CP4	1800.293	1800.193	0.1004
CP5	1812.349	1812.169	0.18073
CP6	2801.855	2802.273	-0.41767
CP7	3198.775	3198.542	0.23293
CP8	3737.522	3737.193	0.32932
RMSE			0.286454

4. 결론

본 연구는 사진측량과 컴퓨터비전간의 내부표정요소에 해당하는 렌즈왜곡 계수를 변환하고 이를 검증하는 연구를 수행하였으며 다음과 같은 결론을 도출하였다.

첫째, 사진측량과 컴퓨터비전간의 내부표정요소 호환성을 위하여 가상의 격자점을 생성하여 입력데이터로 구축하고, 선형방정식을 통한 렌즈왜곡 계수를 변환하는 방법을 제안하였다.

둘째, 렌즈왜곡 변환기법을 위하여 컴퓨터비전 기반의 Pix4D Mapper 소프트웨어와 Matlab에서 제공하는 카메라 캘리브레이션 툴박스를 이용하여 결정된 렌즈왜곡 계수를 사진측량 분야에서 사용하는 렌즈왜곡 계수로 변환하고 이를 검증하였다. 검증결과 변환 오차의 RMSD가 0.5픽셀 이내로 양호하게 도출되었다.

셋째, 컴퓨터비전 기반의 Pix4D Mapper를 이용하여 변환된 렌즈왜곡 계수와 외부표정요소를 이용하여 정밀도화를

위한 사진측량 기반의 에피폴라 영상을 제작하고 검사점의 y-시차를 계산한 결과, 검사점의 RMSE가 0.3픽셀 이내로 양호하게 도출되었다.

본 연구에서는 에피폴라 영상을 이용하여 정밀도화 가능 여부를 간접적으로 실험하였으나 향후 실제 도화장비를 이용한 정밀도화 가능여부와 경제성에 대한 실험이 필요할 것으로 판단된다.

감사의 글

이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2018R1D1A1B07045817).

References

- Bianco, S., Ciocca, G., and Marelli, D. (2018), Evaluating the performance of structure from motion pipelines, *Journal of Imaging*, Vol. 4, No. 98, pp. 1-18.
- Bouguet, J.Y. (2015), Camera calibration toolbox for matlab, *Caltech Vision*, URL: http://www.vision.caltech.edu/bouguetj/calib_doc (last date accessed: 5 August 2019).
- Drap, P. and Lefèvre, J. (2016), An exact formula for calculating inverse radial lens distortions, *Sensors*, Vol. 16, No. 807, pp. 1-18.
- Hartley, R. and Zisserman, A. (2003), *Multiple View Geometry in Computer Vision: 2nd Edition*, Cambridge university press, Cambridge, Cambridgeshire.
- Hong, I.Y. (2016), Image processing for micro UAV with open source software, *Journal of the Korean Cartographic Association*, Vol. 16, No. 3, pp. 139-151. (in Korean with English abstract)
- Hong, S.P., Choi, H.S., and Kim, E.M. (2019), Single photo resection using cosine law and three-dimensional coordinate transformation, *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, Vol. 37, No. 3, pp. 189-198. (in Korean with English abstract)
- Kaehler, A. and Bradski, G. (2016), *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*, O'Reilly Media Inc., Sebastopol, California.
- Kim, E.M., Choi, H.S., and Hong, S.P. (2018). Generation of epipolar image from drone image using direction cosine. *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, Vol. 36, No. 4, pp. 271-277. (in Korean with English abstract)
- Kim, E.M., Choi, H.S., and Park, J.H. (2017), Analysis of applicability of orthophoto using 3d mesh on aerial image with large file size, *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, Vol. 35, No. 3, pp. 155-166. (in Korean with English abstract)
- Kim, T.J., Lim, P.C., Son, J.W., and Seo, S.H. (2019), Feasibility study of 1:1,000 scale map generation using various UAVs and processing SW, *Proceedings of Journal of Korean Society for Geospatial Information System, Korean Society for Geospatial Information Science*, 31-1 May, Busan, Korea, pp. 15-16. (in Korean)
- Lee, J.O., Sung, S.M., and Kim, D.P. (2019), Accuracy assessment of stereo plotting with UAV Images, *Proceedings of Journal of Korean Society for Geospatial Information System, Korean Society for Geospatial Information Science*, 31-1 May, Busan, Korea, pp. 142-143. (in Korean)
- Lim, S.B., Seo, C.W., and Yun, H.C. (2015), Digital map updates with UAV photogrammetric methods, *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, Vol. 33, No. 5, pp. 397-405. (in Korean with English abstract)
- Luhmann, T., Robson, S., Kyle, S., and Harley, I. (2011), *Close Range Photogrammetry Principles, techniques and applications*, Whittles Publishing, Dunbeath, Highland.
- McGlone, J.C. (2013), *Manual of Photogrammetry*. 6th Edition, American Society Photogrammetry and Remote Sensing (ASPRS), Bethesda, MD.
- Mikhail, E.M., Bethel, J.S., and McGlone, J.C. (2001), *Introduction to Modern Photogrammetry*, John Wiley & Sons Inc., New York, N.Y.
- Pix4Dmapper. (2019), How are the internal and external camera parameters defined?, *Pix4D*, URL: <https://support.pix4d.com/hc/en-us/articles/202559089-How-are-the-Internal-and-External-Camera-Parameters-defined> (last date accessed: 27 August 2019).
- Rabiu, L. and Waziri, D.A. (2014), Digital orthophoto

generation with aerial photograph, *Academic Journal of Interdisciplinary Studies*, Vol. 3, No. 7, pp. 133-141.

Verhoeven, G., Doneus, M., Briese, C., and Vermeulen, F. (2012), Mapping by matching: a computer vision based approach to fast and accurate georeferencing of archaeological aerial photographs, *Journal of Archaeological Science*, Vol. 39, No. 7, pp. 2060-2070.