

정사모자이크 제작을 위한 Agisoft Metashape의 병렬처리 성능 평가 Assessment of Parallel Computing Performance of Agisoft Metashape for Orthomosaic Generation

한수희¹⁾ · 홍창기²⁾

Han, Soohee · Hong, Chang-Ki

Abstract

In the present study, we assessed the parallel computing performance of Agisoft Metashape for orthomosaic generation, which can implement aerial triangulation, generate a three-dimensional point cloud, and make an orthomosaic based on SfM (Structure from Motion) technology. Due to the nature of SfM, most of the time is spent on Align photos, which runs as a relative orientation, and Build dense cloud, which generates a three-dimensional point cloud. Metashape can parallelize the two processes by using multi-cores of CPU (Central Processing Unit) and GPU (Graphics Processing Unit). An orthomosaic was created from large UAV (Unmanned Aerial Vehicle) images by six conditions combined by three parallel methods (CPU only, GPU only, and CPU + GPU) and two operating systems (Windows and Linux). To assess the consistency of the results of the conditions, RMSE (Root Mean Square Error) of aerial triangulation was measured using ground control points which were automatically detected on the images without human intervention. The results of orthomosaic generation from 521 UAV images of 42.2 million pixels showed that the combination of CPU and GPU showed the best performance using the present system, and Linux showed better performance than Windows in all conditions. However, the RMSE values of aerial triangulation revealed a slight difference within an error range among the combinations. Therefore, Metashape seems to leave things to be desired so that the consistency is obtained regardless of parallel methods and operating systems.

Keywords : Agisoft Metashape, Parallel Computing, Orthomosaic, GPU, Unmanned Aerial Vehicle

초 록

본 연구에서는 SfM (Structure from Motion) 기술을 기반으로 항공삼각측량을 수행하고 3차원 포인트 클라우드를 생성하며 정사모자이크를 제작할 수 있는 Agisoft Metashape의 병렬처리 성능을 평가하였다. SfM의 속성상 상호표정에 해당하는 Align photos와 3차원 포인트 클라우드를 생성하는 Build dense cloud가 대부분의 시간을 차지하는데, Metashape에서는 이러한 과정에서 CPU (Central Processing Unit)의 다중코어와 함께 GPU (Graphics Processing Unit)를 이용하여 병렬처리를 수행할 수 있다. 세 가지 병렬처리 방법(CPU only, GPU only, CPU + GPU)과 두 가지 운영체제(Windows, Linux)를 조합하여 총 여섯 가지 조건으로 대용량 무인기 영상으로부터 정사모자이크를 제작하였다. 아울러 사용자의 개입 없이 자동화된 방법으로 영상에서 지상기준점을 인식하여 항공삼각측량의 RMSE (Root Mean Square Error)를 측정함으로써 각 조건에 따른 결과의 일관성을 평가하였다. 4220만 화소의 무인기 영상 521장으로부터 정사모자이크를 제작한 결과, 본 연구에서 사용한 시스템에서는 CPU와 GPU의 조합이 가장 나은 성능을 나타내었고 모든 조건에서 Linux가 Windows보다 나은 성능을 나타내었다. 그러나 항공삼각측량의 RMSE를 측정한 결과, 각 설정에 따른 RMSE 값에서 오차 범위 안에서 미세한 차이가 나타났다. 따라서 Metashape는 운영체제 및 병렬처리 여부에 관계없이 동일한 결과가 도출되도록 개선할 여지가 있는 것으로 판단된다.

핵심어 : Agisoft Metashape, 병렬처리, 정사모자이크, GPU, 무인기

Received 2019. 10. 29, Revised 2019. 11. 13, Accepted 2019. 11. 22

1) Member, Dept. of Geoinformatics Engineering, Kyungil University (E-mail: scivile@kiu.kr)

2) Corresponding Author, Member, Dept. of Geoinformatics Engineering, Kyungil University (E-mail: ckhong@kiu.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 무인기(UAV: Unmanned Aerial Vehicle)에서 취득한 고해상도 광학 영상으로부터 대상물에 대한 상세한 정보를 취득하고자 하는 연구가 여러 분야에서 활발히 수행되고 있다. 대표적인 분야로는 광범위 지형 도화(Gao *et al.*, 2019), 송전선/선로 등의 선형 개체 탐사(Zhang *et al.*, 2017; Singh *et al.*, 2017), 건축물의 균열 탐지(Kim *et al.*, 2017)를 들 수 있다. 이러한 목표를 달성하기 위해서는 비행 고도를 낮추어 높은 해상도의 영상을 취득해야 하나 비행 고도를 낮추는 것은 안전상의 문제를 일으킬 수 있고 더불어 제한된 비행 시간 동안 촬영할 수 있는 면적이 줄어드는 단점이 있다. 따라서 적정 비행 고도와 속도, 카메라의 노출 시간 및 구경을 유지하는 비행 설계가 필요하며, 보다 근본적으로는 고해상도의 카메라와 고품질의 렌즈를 사용해야 한다(Han, 2017). 이에 Han and Hong (2017)은 4220만 화소 DSLR (Digital Single Lens Reflex) 카메라와 55mm 단렌즈를 무인기에 장착하여 밀리미터급 GSD (Ground Sampling Distance) 영상을 취득하고 비행 속도와 카메라 설정에 따른 시각적 해상도를 평가하였다. Lee *et al.* (2019)은 1억 화소의 중형 DSLR 카메라를 무인기에 장착하여 초고해상도, 초대용량 광학 영상을 취득하기 위한 연구를 수행하였다. 향후 초고해상도 무인기 영상의 수요 증가와 함께 카메라의 해상도도 지속적으로 향상될 것으로 전망된다.

한편, 항공사진측량의 주요 과정인 카메라 내부표정, 항공삼각측량, 입체도화 등을 수작업으로 진행하던 것과는 달리 최근에는 컴퓨터비전 기술을 이용하여 해당 과정들을 전자동화하고 있다. SfM (Structure from Motion)은 이러한 자동화를 가능하게 하는 대표적인 기술로서, 다양한 사진측량 소프트웨어 및 사진 기반 3차원 스캐닝 소프트웨어³⁾에서 사용하고 있다. 그러나 결과물의 정확성과 신뢰성에 대한 의문이 여전히 존재하며 무인기에서 취득 가능한 영상의 용량도 급격히 증가하면서 처리 시간의 증가도 문제가 되고 있다. Han and Hong (2017)이 취득한 영상의 경우 장 당 120 MB (Megabyte) 내의(raw 형식)를 차지하며 무손실 압축의 경우 80 MB 내의(tiff 형식), 손실 압축의 경우 40 MB 내의(jpeg 형식)를 차지한다. 이러한 영상은 한 회 비행에서도 천 장 내외로 취득할 수 있으며 Agisoft Metashape(이하 Metashape)로 처리할 경우 처리 옵션에 따라 수 시간에서 수 일까지의 시간이 소요된다.

Metashape는 대상물을 다각도에서 촬영한 영상으로부터 3차원 포인트 클라우드 및 메쉬를 자동 생성하는 범용 소프트웨어이다. 뿐만 아니라 카메라 내/외부 표정요소 산출, 지상기준점 입력, DEM (Digital Elevation Model) 생성, 정사영상 및 정사모자이크 생성 등을 자동화하여 항공사진측량 및 도화 분야에도 널리 활용되고 있다. 초기에는 Photoscan이라는 명칭으로 출시하였으나 최근 영상 매칭에서의 노이즈 필터링, GPU (Graphic Process Unit) 가속 개선, cloud 처리 등의 개선을 통해 Metashape로 명칭을 변경하였으며, 세 가지 운영체제(Windows/MacOS/Linux) 버전으로 배포하고 있다. Metashape는 대용량 영상의 처리 시간을 단축하기 위하여 CPU 기반, GPU 기반, network cluster 기반, cloud 기반 병렬처리를 지원한다. 본 연구에서는 Metashape를 이용하여 대용량 무인기 영상으로부터 정사모자이크를 신속하게 생성하기 위한 최적 병렬처리 방법과 운영체제를 제시하는 것을 목표로 한다. 또한 항공삼각측량의 RMSE (Root Mean Square Error)를 계산하여 병렬처리 방법 및 운영체제의 차이에 따른 결과의 일관성을 확인하고자 한다. 이 때 자동화된 방법으로 영상에 지상기준점을 표시하여 사용자에게 의한 오차를 제거하고자 한다. 단, 병렬처리 방법은 CPU only, GPU only, CPU + GPU로 한정하며 network cluster와 cloud 방식은 적용하지 않는다. 운영체제로는 Windows와 Linux를 사용한다.

2. 배경 이론 및 연구 방법

2.1 배경 이론

Agisoft 홈페이지에서는 Metashape가 SfM 기술을 이용하는지 공식적으로 언급하지 않지만 다수의 연구로부터 SfM 기술에 기반한다는 것을 확인할 수 있다(Micheletti *et al.*, 2015; Bianco *et al.*, 2018; Laporte-Fauret *et al.*, 2019; Luhmann *et al.*, 2019). 사용자의 개입이 많이 필요하고 지상기준점을 초반부터 투입하는 전통적인 항공삼각측량과는 달리, SfM 기술은 인접한 영상으로부터 다량의 특징점을 추출하여 영상의 표정 요소 산출 및 3차원 포인트 클라우드 생성을 자동으로 수행하는 것이 특징이다.

SfM의 첫 번째 과정은 각 영상에서 특징점을 추출하는 것이다. 여기에 사용하는 대표적인 방식이 SIFT (Scale Invariant Feature Transform) 객체 인식 기술이다. SIFT는 축척, 회전, 밝기, 관측 위치가 다른 영상으로부터 특징점들을 찾아내기 위

3) Agisoft Metashape, <https://www.agisoft.com/>; Autodesk Recap, <https://www.autodesk.com/products/ Recap/overview>; COLMAP, <https://colmap.github.io/>; OpenMVG, <https://openmvg.readthedocs.io/en/latest/>; Pix4D, <https://www.pix4d.com/>; RealityCapture, <https://www.capturingreality.com/Home>; Regard3D, <http://www.regard3d.org>; VisualSFM, <http://ccwu.me/vsfm/>

한 방법 중 하나이다. SIFT는 특징점인 keypoint를 찾아낸 후 descriptor를 생성하는데, descriptor란 밝기나 관측위치에 영향을 덜 받도록 부분 영상의 gradient를 변환한 것이다(Lowe, 2004). Keypoint의 수는 영상의 질감, 선명도, 해상도에 좌우되는데, 주로 고해상도의 원영상에서 많은 수를 추출할 수 있으며 변화가 적은 표면보다는 질감이 풍부한 대상물에서 보다 좋은 결과를 얻을 수 있다. 성공적인 결과를 위해서는 최소한 세 장의 영상에서 동일 특징점이 관측되어야 하는데, 이는 영상의 조건에 따라 다르며 가끔씩 많은 영상을 취득하는 것이 유리하다 (Westoby *et al.*, 2012).

SfM의 두 번째 과정은 영상의 표정과 dense 포인트 클라우드 생성으로 구성된다. 먼저, 번들 조정을 통하여 각 영상의 표정 요소와 특징점들로 구성된 sparse 포인트 클라우드의 3차원 좌표를 상대적인 값으로 계산한다. 이는 전통적인 항공삼각측량의 상호표정에 해당한다고 볼 수 있다. 이를 위해 approximate nearest neighbour(Arya *et al.*, 1998)와 random sample consensus(Fischler and Bolles, 1987)를 이용하여 서로 다른 영상에 존재하는 동일 특징점쌍을 찾는다. 특징점쌍은 닮음 변환(similarity transformation)에 기반한 관계식에서 영상의 표정 요소를 최소제곱조정으로 산출하기 위한 구속조건으로 사용한다. 이후 공간전방교회를 통해 특징점쌍의 3차원 좌표를 계산한다. 다음으로, 앞에서 산출한 표정 요소를 이용하여 영상쌍에서 중복된 영역을 대상으로 상세한 자동 매칭을 수행한다. 이 과정을 통해 대량의 3차원 포인트를 생성하여 dense 포인트 클라우드를 구성한다.

SfM의 세 번째 과정은 지상기준점을 이용하여 앞에서 생성된 dense 포인트 클라우드의 상대 좌표를 절대좌표로 변환하는 과정, 과오점을 제거하는 과정, 부가적으로 DEM을 생성하는 과정 등으로 구성된다. 일반적으로 절대좌표 변환은 강체 변환(rigid body transformation)을 이용하며 최소한 세 개의 지상기준점을 필요로 한다. Dense 포인트 클라우드는 자동 매칭을 통해 생성되었으므로 과오점이 존재할 가능성이 높으며 과오점의 인식과 제거는 수동으로 하거나 별도의 알고리즘을 적용해야 한다.

2.2 연구 방법

Metashape은 2.1에서 설명한 SfM 과정과 유사한 형태로 구성된 workflow 메뉴를 따라 영상을 처리한다. 단, 2.1에서는 세 번째 과정에서 지상기준점을 입력하여 절대좌표 변환을 수행하였으나, Metashape에서는 두 번째 과정 중 영상의 표정과 dense 포인트 클라우드 생성 과정 사이에 지상기준점 정보를 입력한다. Metashape에서 정사모자이크를 제작하기 위한 최소 과정은 Table 1과 같다.

Table 1. Minimal workflow to generate an orthomosaic in Metashape

Workflow	Description
Add photos	Image loading
Align photos	Relative orientation and sparse point cloud generation
GCP (Ground Control Points) marking (not in the workflow menu)	Manually marking GCP on the screen
Optimize cameras (in tools menu)	Absolute orientation with IOP (Interior Orientation Parameter) update
Build dense cloud	Dense point cloud generation
Build DEM	DEM generation
Build orthomosaic	Orthomosaic generation

Table 1에서 align photos와 build mesh 과정은 2.1의 첫 번째와 두 번째 과정에 해당하는 특징점 추출, 동일 특징점쌍 탐색, 자동 매칭으로 구성되며 전 과정 중에서 가장 많은 시간을 소요한다. 한편, Metashape는 대부분의 과정에서 처리 속도를 향상시키기 위하여 CPU의 다중 코어를 사용하여 병렬처리를 수행하는데, align photos와 build mesh 과정에서는 추가로 GPU를 사용하여 병렬처리를 수행할 수 있다. Build mesh나 build texture 과정에서도 GPU를 사용하여 병렬처리를 수행할 수 있으나 본 연구에서는 논외로 한다.

Metashape의 병렬처리 성능과 결과물의 일관성을 평가하기 위하여, GPU 사용 여부에 따른 처리 시간을 비교하고 지상기준점을 사용하여 항공삼각측량의 RMSE를 측정한다. 다만, 사용자가 지상기준점을 입력하는 과정에서 정확도의 차이가 발생할 수 있으므로 지상기준점 입력은 자동화된 방법을 사용한다. 이를 위하여 Metashape에서 제공하는 특별한 형식의 대공 표지(Fig. 1)를 출력하여(tools - markers - print markers) 촬영 대상 지역에 설치하고, 대공 표지의 자동 인식 및 중심점 디타이핑도 Metashape의 자동 인식 기능을 이용한다(tools - markers - detect markers). 본 연구에서 정사모자이크를 제작하기 위해 제안하는 과정은 Table 2와 같다.

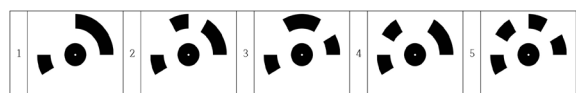


Fig. 1. Aerial targets detectable in Metashape (Han and Hong, 2017)

Table 2. Alternative workflow proposed by the present study

Workflow	Description
Add photos	Image loading
Detect markers (in tools menu)	Automatic GCP recognition
Align photos	Relative and absolute orientation, and sparse point cloud generation
Optimize cameras (in tools menu)	IOP update
Build dense cloud	Dense point cloud generation
Build DEM	DEM generation
Build orthomosaic	Orthomosaic generation

3. 실험 및 결과

3.1 자료와 시스템의 제원

본 연구에서 사용한 무인기와 처리 시스템의 제원은 Tables

3 and 4와 같다. 무인기는 DJI Matrice 600으로서 Ronix MX 짐벌을 탑재하고 있으며 최대 적재 중량은 6kg, 최대 적재 시 비행 시간은 15분 정도이다. 카메라는 35mm CMOS를 탑재한 미러리스 카메라로서 약 4220만 화소의 영상을 취득할 수 있으며, 렌즈는 단렌즈로서 제조사에서 발표한 초점거리는 55mm이다. 처리는 동일한 컴퓨터에서 Windows와 Linux로 부팅하여 Windows 버전과 Linux 버전 Metashape를 사용하여 수행하였다.

정사모자이크 제작에 사용한 영상의 제원과 영상 취득과 관련된 설정 내역은 Table 5와 같다. 모든 영상은 raw 형식으로 촬영하여 비손실 압축 tiff로 변환하였으며, F-stop 값은 4.5, 노출 시간은 1/1000초, ISO는 100으로 고정하였다. 무인기는 이륙 위치로부터 약 83m 고도에서 최대 속도 5 m/s로 비행하였으며, 약 15분간 1000매 이상의 영상을 촬영하여 그 중 지상기준점 인근의 영상 541매를 선택하였다.

지상기준점은 Fig. 1과 같은 형태의 대공 표지를 60 x 60 cm 크기로 출력하여 주로 건물의 옥상에 배치하였으며, 토탈 스테

Table 3. Specifications of UAV

UAV		Camera		Lens	
Model	DJI Matrice 600	Model	Sony ILCE-7RM2	Model	Sony SEL55F18Z
Wing type	6 rotary wings	Type	Lens exchangeable mirrorless	Type	Prime lens
Weight(max.)	9.5 kg(15.5 kg)	Weight	625 g	Weight	281 g
Flight time	32 min (16 min in 15.5 kg)	Pixels	42.2 mega	Focal length	55 mm
Gimbal	Ronin MX	Sensor type	Full frame (35 mm) CMOS	Max. aperture	f/1.8

Table 4. Specifications of system and software

Configuration	1	2
CPU	AMD Threadripper 2970WX 24 core processor 3.00 GHz	
RAM	PC4-21300 64 GB	
SSD	M.2 1 TB	
GPU	NVIDIA GTX 1060 3 GB 1152 cores	
OS	Windows 10 64 bit	Ubuntu desktop 18.04
Metashape version	1.5.2 for windows	1.5.2 for linux

Table 5. Specifications of image and flight

Image		Image acquisition plan	
Dimension	7952 x 5304 pixels, 8bit RGB	Location	Hayang-Eup, Gyeongsan-Si, Gyeongbuk, South Korea
Format	tiff (LZW lossless compression)	Avg. flight altitude and GSD	82.7 m (estimated) 6.7 mm/pixel (estimated)
EXIF	f/4.5, 1/1000 sec, ISO 100	Max. flight speed	5 m/s
Capacity	53.77 MB - 80.76 MB	Coverage	26200 m ²

Table 6. 3D coordinates of GCPs (unit: meter)

Point ID	x	y	z	Point ID	x	y	z
target 6	12.453	-42.509	-13.393	target 7	64.580	-7.763	0.000
target 5	-31.520	-61.692	-12.717	target 1	0.000	0.000	0.000
target 131	57.242	-39.387	-3.138	target 3	66.57	34.377	2.871
target 4	19.183	-33.722	-3.100	target 2	26.297	36.685	2.903

이션을 이용하여 3차원 상대 좌표를 취득하였다. 지상기준점의 배치 형태와 좌표는 각각 Fig. 2와 Table 6과 같다.



Fig. 2. Placement of GCPs

3.2 실험 방법

Metashape에서는 GPU의 사용 여부를 선택하는 옵션(tools - preferences - GPU, 이후 GPU 옵션)과 함께 “Use CPU when performing GPU accelerated processing”이라는 보조 옵션(이후 CPU 옵션)을 제공한다. GPU 옵션을 활성화하면 align photos, build dense cloud의 과정에는 CPU를 사용하지 않고 GPU만을 사용하여 병렬처리하고, 비활성화하면 CPU만을 사용하여 병렬처리한다. GPU 옵션을 활성화한 경우에는 사용하는 GPU의 개수만큼 CPU의 코어를 점유하는데, 예를 들어, 두 개의 GPU를 사용한다면 두 개의 CPU 코어를 점유한다.⁴⁾ CPU

옵션을 활성화하면 GPU에 점유되지 않은 나머지 CPU 코어도 align photos, build dense cloud 과정의 병렬처리에 투입한다. 일반적으로 고성능의 CPU와 저성능의 GPU를 사용할 때에는 CPU 옵션을 활성화하는 것이 유리하고, 고성능의 GPU를 사용할 때에는 CPU 옵션을 비활성화하는 것이 유리하다고 알려져 있다.⁵⁾ 그림에도 불구하고 CPU 옵션의 활성/비활성에 따른 병렬처리 성능의 향상 여부는 실험을 통해서 검증해야 한다. CPU의 코어가 영상 처리를 담당하는 것과 GPU의 컨트롤을 담당하는 것 중 어느 것이 효율적인가를 사용자가 정확히 판단하기 어렵기 때문이다.

본 연구에서는 CPU only, GPU only, CPU + GPU의 병렬처리 방법과 Windows, Linux의 운영체제를 조합하여 총 여섯 가지 조건으로 정사모자이크를 제작하였다. 각 조건에 의한 항공삼각측량 결과의 RMSE 평가는 Table 6의 모든 지상기준점을 이용하여 수행하였다. 일반적으로 지상기준점과 검사점을 구분하여 지상기준점으로는 항공삼각측량을 수행하고 검사점으로는 결과의 정확도를 평가한다. 그러나 본 연구에서는 병렬처리 방법과 운영체제에 따른 일관성 검증을 목표로 하므로 지상기준점과 검사점을 구분하지 않고 RMSE를 평가하였다.

3.3 실험 결과

Table 2의 정사모자이크 제작 과정 중 첫 번째 과정과 두 번째 과정(영상입력, 대공 표지 입력, 지상기준점 좌표 입력)을 제외한 나머지 과정을 batch process로 진행하면서 각 과정에 소요된 시간을 측정하였다. Align photos는 accuracy를, build dense cloud는 quality를 선택할 수 있는데, 실험에 지나친 시간이 소요되지 않도록 두 과정 모두 medium quality를 선택하였으며, 나머지 과정에서는 기본 옵션을 선택하였다. 영상입력으로부터 정사모자이크 제작까지의 각 과정마다 도출된 결과를 확인하기 위하여 리포트를 생성하였다(file - export - generate report). 각 과정별 결과는 Table 7과 같다.

4) Agisoft PhotoScan Multi Core Performance, <https://www.pugetsystems.com/labs/articles/Agisoft-PhotoScan-Multi-Core-Performance-709/>; Agisoft PhotoScan GPU Acceleration, <https://www.pugetsystems.com/labs/articles/Agisoft-PhotoScan-GPU-Acceleration-710/>

5) Agisoft PhotoScan 1.4.1 - CPU and GPU Preferences, <https://www.pugetsystems.com/labs/articles/Agisoft-PhotoScan-1-4-1---CPU-and-GPU-Preferences-1149/>

Table 7. Processing results

Workflow	Item	Windows 10			Ubuntu desktop 18.04		
		CPU only	GPU only	CPU+GPU	CPU only	GPU only	CPU+GPU
Align photos	Time	32 min 28 sec	22 min 40 sec	15 min 57 sec	31 min 55 sec	22 min 2 sec	14 min 54 sec
	Speed up	N/A	1.43	2.04	N/A	1.45	2.14
	No. of points	90410	89907	89571	89770	89992	89566
Detect markers	Time	Approximately 90sec + manual coordinate input					
Optimize camera	Time	11 sec	14 sec	12 sec	15 sec	11 sec	13 sec
Build dense cloud	Time	32 hour 26 min	6 hour 40 min	5 hour 58 min	20 hour 9 min	5 hour 49 min	5 hour 4 min
	Speed up	N/A	4.87	5.45	N/A	3.46	3.98
	No. of points	67115619	67478846	67410617	69682240	70034171	69986592
Build DEM	Time	50 sec	49 sec	49 sec	27 sec	27 sec	26 sec
	Dimension	8592 x 9826	8676 x 9927	8698 x 9930	8623 x 9889	8686 x 9941	8705 x 9942
Build orthomosaic	Time	12 min 46 sec	12 min 38 sec	12 min 42 sec	9 min 9 sec	9 min 1 sec	9 min 5 sec
	Dimension	25321 x 30368	25348 x 30384	25315 x 30371	25329 x 30386	25327 x 30384	25326 x 30383
Overall time		33 hour 12 min	7 hour 16 min	6 hour 28 min	20 hour 51 min	6 hour 21 min	5 hour 29 min
RMSE (cm)	X	1.26519	1.26507	1.26822	1.26313	1.26577	1.26313
	Y	1.16231	1.16887	1.17755	1.17229	1.16964	1.16037
	Z	0.98240	0.98476	0.99839	0.98025	0.98405	0.99651

운영체제에 따른 비교에서는 Linux가 Windows에 비하여 대체로 나은 성능을 나타내었다. 가자 많은 시간이 소요되는 build dense cloud 과정에서, CPU only 방법은 Linux가 Windows에 비해 약 1.6배의 성능을 나타내었고, GPU only 또는 GPU + CPU 방법은 근소한 차이로 나은 성능을 나타내었다. 그 다음으로 많은 시간이 소요되는 align photos 과정에서는 Linux가 근소한 차이로 나은 성능을 나타내었다. Build DEM이나 build orthomosaic 과정에서는 보다 큰 차이를 나타내었으나 해당 과정의 처리 시간이 차지하는 비중은 높지 않았다.

GPU 사용 여부에 따른 비교에서는 GPU를 사용하는 방법이 CPU만을 사용하는 방법에 비하여 월등한 성능을 나타내었다. Build dense cloud 과정에서는 Windows의 경우 CPU only 방법에 비하여 GPU only 방법이 4.87배, GPU + CPU 방법은 5.45배의 성능 향상을 나타냈으며, Linux의 경우도 각 방법에 대하여 3-4배의 성능 향상을 나타냈다. Align photos 과정에서도 Windows의 경우 CPU only 방법에 비하여 GPU only 방법은 1.43배, GPU + CPU 방법은 2.04배의 성능 향상을 나타냈으며 Linux의 경우도 유사한 성능 향상을 나타냈다. 다만, 3.2에서 언급했듯이, GPU only 방법과 GPU + CPU 방법의 성능은 CPU와 GPU의 성능, GPU의 개수에 따라 우열이 바뀔 수 있으므로, 본 연구에서와 같이 GPU + CPU 방법이 항상 나은 성능을 나

타낸다고는 말할 수 없다.

지상기준점을 이용한 항공삼각측량 결과의 RMSE 비교에서는 다소의 문제점을 발견할 수 있었다. 여섯 가지 조건에서 측정된 RMSE를 비교하면 센티미터 단위로 소수점 이하 두 자리부터 서로 다르다는 것을 확인할 수 있다. 이는 미터 단위로 입력한 지상기준점의 유효 숫자 이하의 차이로서 실질적인 정확도에는 차이가 없다고 보아도 무방하다. 그러나 병렬처리 방법과 운영체제에 관계없이 동일한 결과가 도출되어야 하는 기본 원칙에는 위배된다고 볼 수 있다. 심지어 같은 병렬처리 방법과 운영체제에서 처리를 반복해도 서로 다른 RMSE가 산출되었다. 이러한 현상의 원인으로는, align photos의 결과인 sparse 포인트 클라우드(점합점)의 포인트 수와 내부 표정요소(본 논문에는 제시하지 않음)가 서로 다름에서 찾을 수 있다. 보다 근본적으로는 소프트웨어의 설계 및 구현에 보다 세심한 주의가 필요했던 것으로 판단할 수 있다. 결과로는 dense 포인트 클라우드의 포인트 수, DEM과 정사모자이크의 크기도 조금씩 다르게 나타났다.

4. 결론

본 연구에서는 SfM을 기반으로 항공삼각측량 및 3차원 포

인트 클라우드를 자동으로 생성하는 Metashape를 이용하여 대용량 무인기 영상으로부터 정사모자이크를 제작하기 위한 병렬처리 성능을 평가하였다. SfM의 속성상 상호표정에 해당하는 align photos와 3차원 포인트 클라우드를 생성하는 build dense cloud가 대부분의 시간을 소요하는데, Metashape에서는 이러한 과정에서 CPU의 다중코어뿐만 아니라 GPU를 이용하여 병렬처리를 수행할 수 있다. CPU only, GPU only, CPU + GPU의 병렬처리 방법과 Windows와 Linux의 운영체제를 조합한 여섯 가지 조건에서 정사모자이크를 제작한 결과, Linux에서 Windows보다 대체로 나은 성능을 나타내었고 GPU를 사용하는 방법에서 CPU만을 사용하는 방법보다 월등한 성능을 나타내었다. 따라서 GPU를 사용하여 병렬처리를 수행하는 것이 바람직하다고 볼 수 있다. GPU를 사용한다면 Linux와 Windows의 성능 차이는 현저하게 줄어들므로 운영체제의 선택은 크게 중요하지 않은 것으로 판단된다. 그러나 지상기준점을 이용하여 항공삼각측량의 RMSE를 비교한 결과, 각 조건에 따른 RMSE 값에서 오차 범위 안의 미세한 차이가 나타났다. 따라서 Metashape는 운영체제 및 병렬처리 여부에 관계없이 동일한 결과가 도출되도록 개선할 여지가 있는 것으로 판단된다.

향후 연구에서는 network cluster와 cloud를 이용한 병렬처리 성능 개선 효과를 검증하고자 한다. 아울러 향후 출시되는 버전의 Metashape에서는 운영체제 및 병렬처리 여부에 상관없이 일관된 결과가 도출되는지 검증하고자 한다.

감사의 글

본 과제는 행정안전부 재난안전 산업육성지원 사업의 지원을 받아 수행된 연구임(2019-MOIS32-015).

References

- Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., and Wu, A.Y. (2007), An optimal algorithm for approximate nearest neighbor searching fixed dimensions, *Journal of the ACM*, Vol. 45, No. 6, pp. 891–923.
- Bianco, S., Ciocca, G., and Marelli, D. (2018), Evaluating the performance of structure from motion pipelines, *Journal of Imaging*, Vol. 4, No. 8:98.
- Fischler, M.A. and Bolles, R.C. (1981), Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395.
- Gao, M., Xu, X., Klinger, Y., Woerd, J. van der, and Taponnier, P. (2017), High-resolution mapping based on an unmanned aerial vehicle (UAV) to capture paleoseismic offsets along the Altyn-Tagh fault, China, *Scientific Reports*, Vol. 7:8281.
- Han, S. (2017), High-resolution and high-definition image acquisition using UAV and high-precision aerial triangulation, *Journal of Korean Society for Geospatial Information Science*, Vol. 25, No. 3, pp. 101-109. (in Korean with English abstract)
- Han, S. and Hong, C.-K. (2017), Acquisition of subcentimeter GSD images using UAV and analysis of visual resolution, *Journal of the Korean Society of Survey, Geodesy, Photogrammetry, and Cartography*, Vol. 35, No. 6, pp. 563-572. (in Korean with English abstract)
- Kim, H., Lee, J., Ahn, E., Cho, S., Shin, M., and Sim, S.-H. (2017), Concrete crack identification using a UAV incorporating hybrid image processing, *Sensors*, Vol. 17, No. 9:2052.
- Laporte-Fauret, Q., Marieu, V., Castelle, B., Michalet, R., Bujan, S., and Rosebery, D. (2019), Low-cost UAV for high-resolution and large-scale coastal dune change monitoring using photogrammetry, *Journal of Marine Science and Engineering*, Vol. 7, No. 3:63.
- Lee, J., Choi, H., and Kim, D. (2019), Accuracy evaluation of stereo plotting with medium format camera image acquired by a drone, *Fall Conference of Korean Society for Geospatial Information Science*, 31 May–1 June, Busan, South Korea. (in Korean)
- Lowe, D.G. (2004), Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, Vol. 60, No. 2, pp. 91–110.
- Luhmann, T., Chizhova, M., Gorkovchuk, D., Hastedt, H., Chachava, N., and Lekveishvili, N. (2019), Combination of terrestrial laserscanning, UAV and close-range photogrammetry for 3D reconstruction of complex churches in Georgia, *2nd International Conference of Geomatics and Restoration*, 8-10 May, Milan, Italy, pp. 753–761.
- Micheletti, N., Chandler, J., and Lane, S.N. (2015), Structure from motion (SfM) photogrammetry, In: Clarke, L.E. and Nield, J.M. (Eds.), *Geomorphological Techniques (Online Edition)*, British Society for Geomorphology, London, Chap. 2, Sec. 2.2.
- Singh, A.K., Swarup, A., Agarwal, A., and Singh, D. (2017), Vision based rail track extraction and monitoring through

drone imagery, *ICT Express*, In press.

Westoby, M.J., Brasington, J., Glasser, N.F., Hambrey, M.J., and Reynolds, J.M. (2012), 'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications, *Geomorphology*, Vol. 179, pp. 300–314.

Zhang, Y., Yuan, X., Li, W., and Chen, S. (2019), Automatic power line inspection using UAV images, *Remote Sensing*, Vol. 9, No. 8:824.