MULTI-VIEW-STEREO DSM FROM PLANETSCOPE SATELLITES

COMPUTER VISION PROCESS WITH REDUNDANT FRAME-BASED SATELLITES

Author

Valentin Schmitt (Planet intern, University of Stuttgart, IfP)

Supervisor

Apl. Prof. Dr.-Ing. Norbert Haala (University of Stuttgart, IfP)

Stakeholders

Product Engineering: Kelsey J., Matthias K., Seth P., Matthias K., Duy N., Brian B.
Mission Ops: Jose G

TABLE OF CONTENTS

FIGURES

TABLES

EQUATIONS

ABSTRACT

Since 2015, Planet Labs has operated Cubesat 3U satellites for Earth observation and remote sensing monitoring. The Planetscope constellation contains around 130 Doves which are frame based sensors and they reached the daily time resolution in 2018. The whole setting is close to airborne photogrammetric acquisitions that allow digital surface modelling computation with Structure from Motion pipelines. Unlike low flight, Doves perch on orbits around 475 km high with a small field view angle acquiring 24 km x 16 km scenes and thus they hold a weak intersection geometry at the ground. Nevertheless, they provide a very large redundancy. This study aims to design an automated process chain for 3D reconstruction enhanced with information redundancy. Product knowledge of overall organisation, hardware and software description as well as internal process specification outlines first hassles and sets a coarse methodology forth. Former studies about Planetscope matching built object oriented methodology which returned promising results. However, this research focuses on an image oriented approach which is closer to low flight custom. The research part sets the whole chain up which is divided into scene block creation, bundle adjustment and reconstruction sections. From the entire Planetscope database, the scene selection retrieves the best ones and sets a working directory in. Then, the adjustment section presents the strength and weakness of incoming Dove's location models and it figures out a correcting procedure making use of the SRTM only. The image matching provides stereo point clouds with a large redundancy and the reconstruction part ends with a multi-view stereo method merging elevations and returns a standard DSM and its computation accuracy. External data (SRTM) is only involved during the bundle adjustment and reconstruction results are compared to ground truth without further transformation. That assessment runs over test sites (Providence Mountain, Stuttgart and Mount Saint Helens) and the last part outlines process issues and summarises the achievable accuracy. The final accuracy reaches 8 m in vertical direction (2 pixels) along a 4 m horizontal grid. All in all, this research sets the basis of Planetscope reconstruction with image approach, even though some computation limits remain and it covers only the case of Dove-Classic. Hence, all process improvements and extension with following satellite generation shall again improve outcomes. Furthermore, the final algorithm is almost automatic and it can be implemented on Planet customer portal.

INTRODUCTION

This is the thesis report of the Master of Science program achieved by Valentin Schmitt from University of Stuttgart. It deals with DSM computation from Planetscope and its achievable accuracy. It wraps up the research parts during an inside Planet labs experience (internship) and the result assessment.

Planet Labs inc. (Planet) is an American company established in 2010 managing Earth observation satellites. It currently manages and releases Rapideye, Skysat and Planetscope acquisition products. That thesis study conveys on this last satellite constellation also known as Doves, which is the company speciality. Planetscope (PS) scenes of 4 m GSD are in-use for remote sensing purposes but its geometry reminds us of a traditional airborne photogrammetric acquisition. After prior study, it became clear that the PS setting provides a sufficient geometry Structure from Motion chain (SfM) as long as we take care over some key parts. From several perspectives, sensed with frame-based cameras, the traditional photogrammetry identifies homologs, matches them, intersects these rays, generates point clouds and derives a DSM. PS flights also provide those perspective differences in overlapping areas. However, the geometry setting does not provide a clear intersection angle unlike airborne acquisition. The Base-Height ratio is much smaller than airborne acquisition due to flying height and thus the incidence angle is inaccurate. Nevertheless, PS senses the entire Earth surface everyday. That short revisit time from slightly different positions gives an impressive redundancy.

Former studies about satellite dense matching, described hereafter, showed the PS capability and this current research comes as an extension to them. It starts with an image based approach and it brings several new search axes in matters of computer vision knowledge and redundancy enhancement. The outcome gathers workflow design, its implementation and its assessment at a time. Phrased as a question, that research shall answer **“what is the reachable accuracy of Planetscope reconstruction from image based approach using redundancy enhancement?”**

PRIOR RESEARCH PRESENTATION

DEM Generation from Multi Satellite PlanetScope Imagery (Ghuffar *et al.*, 2018)

Ghuffar (2018) designed a DSM production chain from the same constellation over several test sites. It proves that a right process exists to extract a consistent DSM with an object based approach. It starts with a bundle adjustment solving an affine transformation relative to Rational Polynomial Coefficient (RPC) to guarantee the image alignment. It is an essential step for the following correlation which is discussed in our Absolute Structure from Motion section 2.2. Then, a matching cost algorithm in 3D space selects the best stereo pair which is matched by Semi Global Matching (SGM). Finally, a 3D transformation brings the DSM closer to its reference. That least square adjustment moves back the product which was shifted during the relative RPC adjustment. His

article presents mainly cross orbit matching due to the matching cost algorithm. It comes up with promising altitude standard deviation, NMAD: 2.80 to 8.91 m

Automated DEM generation from very high resolution Planet SkySat (Bhushan *et al.*, 2021)

During an internship at Planet in 2020, Burshan (2021) led a study about DEM creation from SkySat. That high resolution satellite uses a similar pattern than Planetscope with a push-frame sensor but the 3.5 m focal length yields 50 cm GSD images. Skysat acquisitions are tasked by the operation team and the stereo acquisition comes from forward and backward off-targeting during the flight. Mr Burshan figured out the process chain and reached an interesting accuracy: NMAD: 0.73 to 3.59 m. He also made available in the Planet IT system several interesting tools like Ames Stereo Pipeline that we use too for several reasons described below.

Optimization of Optical Image Geometric Modelling, Application to Topography Extraction and Topographic Change Measurements Using PlanetScope and SkySat Imagery (Aati *et al.*, 2020)

Aati *et al.* (2020) released a study about dense matching from Skysat and Planetscope. He also adjusted RPCs models, called Rational Functional Model (RFM), by affine transformation. He built and implemented that adjustment function in the open-source project COSI-Corr, led by the California Institute of Technology (CalTech). Then, he runs the dense matching from the object space approach, similar to Ghuffar (2018). He reached an accurate product that can be used for glacier or earthquake monitoring: NMAD: 9.35 to 12.34 m.

Other publications about optical satellite dense matching exist but these 3 ones appeared to be the most relevant ones according to our subject. More specific publications are also quoted after as reference to precise issues.

STUDY CONTENT

Our first section 1 describes the Planetscope setting. The position as Planeter gave us access to internal information about the satellite manufacturing and internal processes (Planet Pipeline). A clear understanding of input data shall present forthcoming issues and challenges. The PS setting has evolved along years of development with several satellite generations (hardware) and the name Planetscope appoints all these versions and their respective process method.

Then, we implement our processing method based on the 3 scientists quoted before and we increase them with additional development leading to a final algorithm. The main difference is the image based approach. It also aims to be an autonomous user tool which shall be able to run from user specification (AOI, dates) and yields one or several DSM outputs. The whole chain has been divided into 3 main steps with nicknames for later referencing, set forth in the Development and Implementation section 2. Firstly, we have to extract image blocks from the very large database of PS

scenes, named Stereo-Scene Block Parsing (SSBP): section 2.1. Then, a bundle adjustment corrects misalignment between scenes ensuring relative position from image to image, named Absolute Structure from Motion (ASfM): section 2.2. Finally, the dense matching part reconstructs a 3D geometry from the acquisition and enhances the accuracy by point redundancy, named Multi-Scene Stereo (MSS): section 2.3.

At the end, we run the outcoming algorithm over 3 test sites with different characteristics in order to state the achievable results, the accuracy and difficulties according to each test site. This is discussed in the last part Results: section 3.

The development was organised with the SCRUM-AGILE principles of project management. Instead of a linear development, step after step, we follow development loops which return a new brick version. One loop relies on Dove generations with hardware and software differences and another runs through the 3 algorithm steps quoted before. Hence, we cover the step loop with the first generation and forthcoming challenges with other generations are outlined in this report. Each Dove generation run faces new issues and has to include solutions in the algorithm. This report presents the full algorithm and solutions developed for Dove-Classic.

The algorithm is written in Python and built like a library. The convenience of Python code, coupled with additional libraries like OpenCV, Rasterio, Numpy, etc, allows us to implement some key functions, called handmade. It is also a convenient way to stack external processors from libraries like Gdal (C++ executable) used for standard image processing. The organisation of the repository was fixed at the early stage with main scripts and modules of functions. The implementation of computer vision processes is done with Ames Stereo Pipeline (ASP). It is an open-source project led by NASA engineers. They have built a library written in C++ wrapping OpenCV, Gdal, Census and other libraries. It has been designed for satellite image matching and thus avoids some complications like coordinate systems because it runs within ECEF cartesian frame. These facilities were highlighted by Mr Bhushan and he set ASP in the Planet system during his internship. All in all, one can run our algorithm over a new area by simply calling the main script which runs the whole process chain.

Beside, we select 3 Area Of Interest (AOI) to design and run the final algorithm. We also need a ground truth per AOI which has to fulfil some criteria. The area must have a consistent size because a small area limits processing time but a large one makes use of block creation algorithms. Due to Planet Labs organisation, we would like to select one test site in the US and one in Europe. The Multi-View Stereo (MVS) algorithm returns point clouds and its gridding is related to data merging. So, the reference data would be preferably a point cloud but a rasterized file is a lighter product as long as the GSD is a round number which can be easily subsampled. The ground truth has to be 1/10 of the expected accuracy. We expect an accuracy box of 1 time the original Planetscope GSD meaning 4m side and the reference has to be 40cm accuracy. The reference data must include vegetation in order to be compared with optical photogrammetry products. The area should be reasonably stable in terms of motion (earthquake) and vegetation allowing cross-season merge research. Hence, we came up with the following test sites and ground truth.

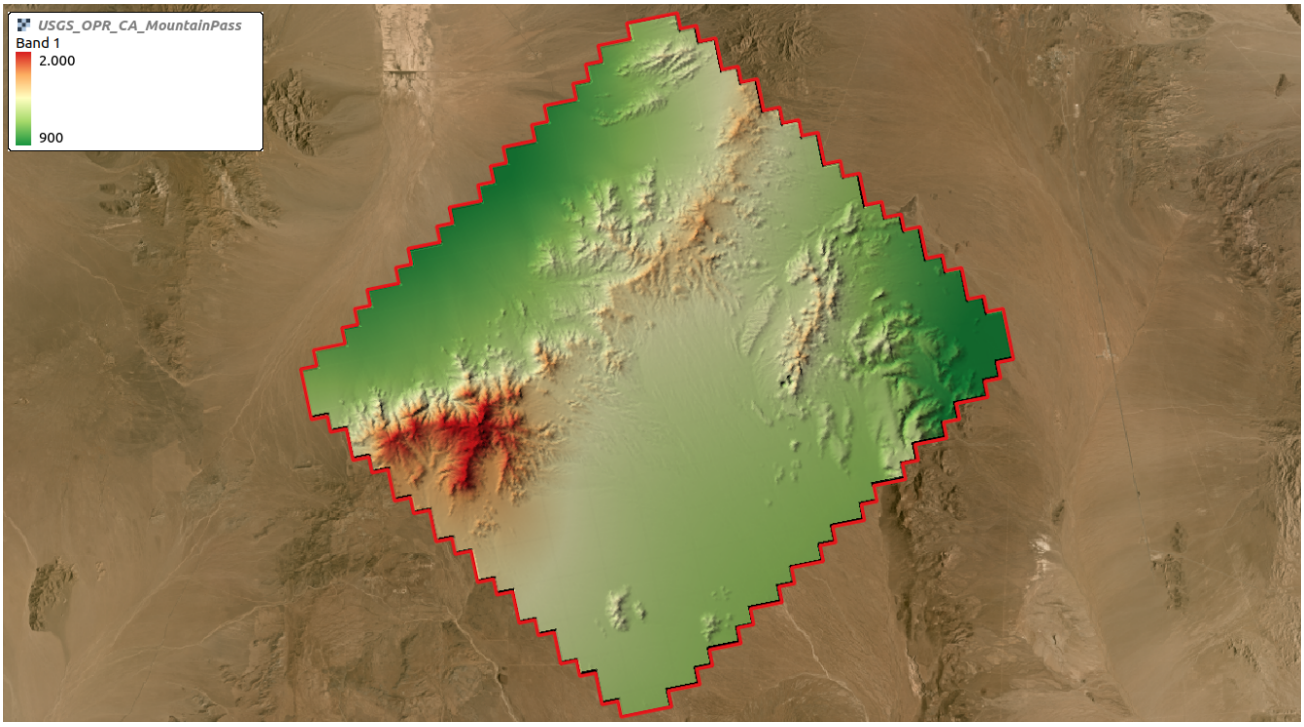
Providence Mountain, USA (California)

Providence mountain is a mountainous desert near Las Vegas in Mojave national park. This is the development test site with a very low cloud cover providing many clear scenes. It also has a large height range. The ground truth is a LiDAR scan collected by USGS released on Opentopography in raster version.

Table 1: Providence Mountain ground truth acquisition parameters

PROVIDENCE MOUNTAIN REFERENCE DATA, PARAMETERS			
LiDAR parameters	LiDAR values	DSM parameters	DSM values
Survey periode	August to October 2019	Data format	Tif DSM (GSD: 1 m)
Resolution	6.31 points / m^2	Position reference	NAD83(2011) / Conus Albers (EPSG: 6350)
Accuracy	QL2: Vertical: 10 cm (95^{th} percentile)	Altitude system	NAVD88 height (assumed EGM96)
Area / Height range	1 500 km^2 / 900 - 2300 m	Bit depth	Float 32 bits

Figure 1: Providence Mountain reference DSM



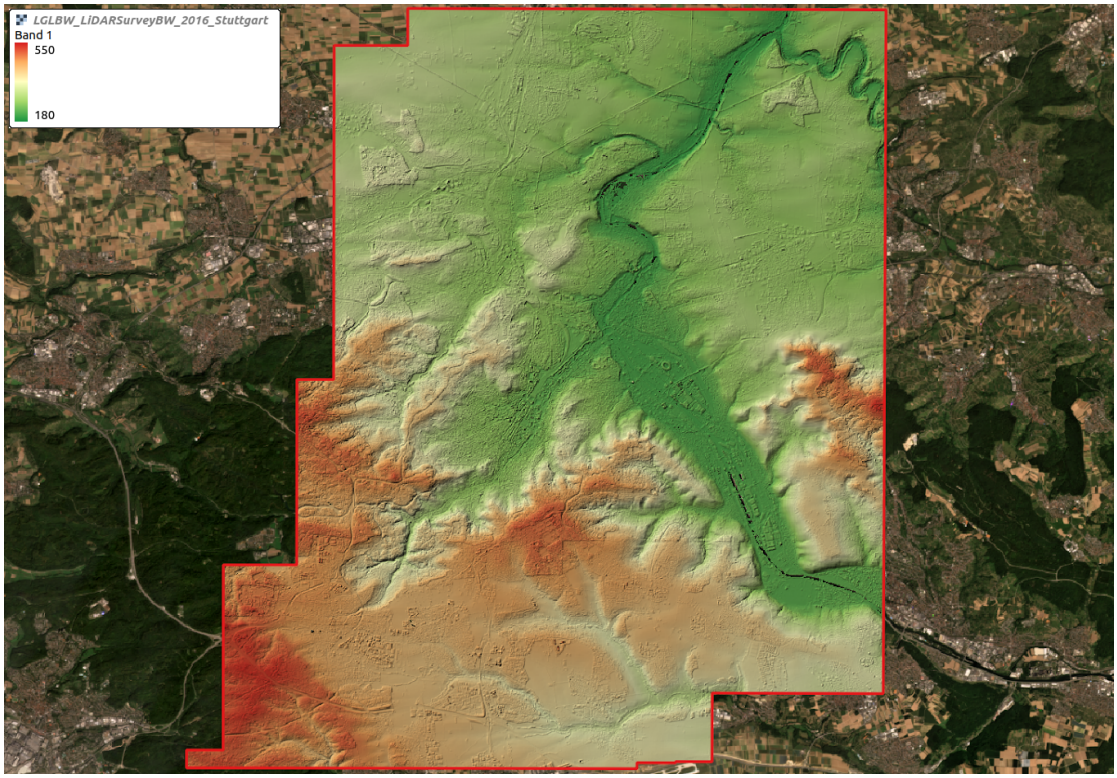
Stuttgart, Germany

The city of Stuttgart is the second test site. It is an urban area with a smaller height range and agricultural fields. We are interested in measuring the effect of vegetation changes and the building detectability. The ground truth is a LiDAR acquisition from the state survey institute shared by the Institut für Photogrammetry (IfP) in raster version.

Table 2: Stuttgart ground truth acquisition parameters

STUTTGART REFERENCE DATA, PARAMETERS			
LiDAR parameters	LiDAR values	DSM parameters	DSM values
Survey periode	Between 2016 and 2022	Data format	Tif (0.5 m)
Resolution	8 points / m²	Position reference	ETRS89 / UTM (EPSG Code 25832)
Accuracy	Horiz: +/- 30 cm Vert: +/- 20 cm	altitude system	NHN (DHHN2016 with altitude status 170)
Area / Height range	700 km² / 180 - 550m	Bit depth	Float 32 bits

Figure 2: Stuttgart reference DSM



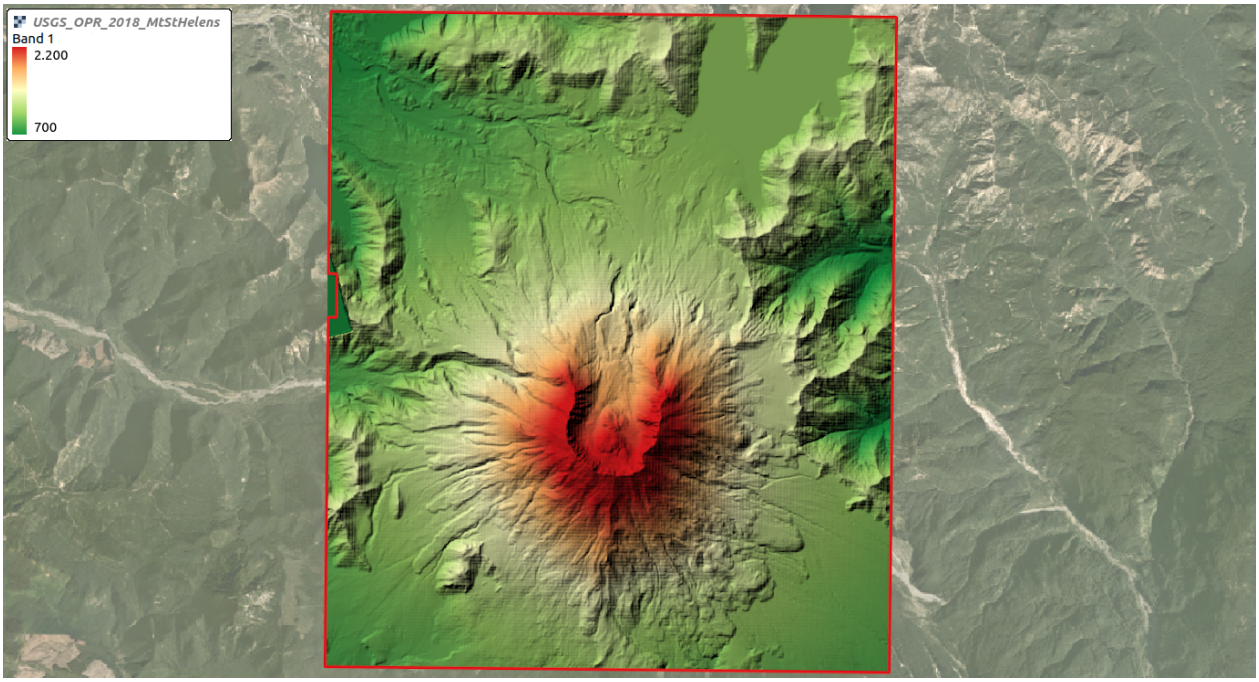
Mount St Helens, USA (Washington)

The last test site is Mount St. Helens extinct volcano. That was Mr Bhushan's test site and it is worth comparing the outcoming results with Skysat matching even though it varies along the year with the snow cover and vegetation changes. The ground truth is another USGS LiDAR acquisition released on Opentopography.

Table 3: Mount Saint Helens ground truth acquisition parameters

MOUNT SAINT HELENS REFERENCE DATA, PARAMETERS			
LiDAR parameters	LiDAR values	DSM parameters	DSM values
Survey periode	August 2018 to April 2019	Data format	Tif (GSD: 1 m)
Resolution		Position reference	NAD83(2011) / UTM zone 10N (EPSG:6339)
Accuracy	Vert: 12 cm (95th percentile)	altitude system	NAVD88 (GEOID12B) (assumed EGM96)
Area / Height range	400 km^2 / 700 - 2200m	Bit depth	Float 32 bits

Figure 3: Mount Saint Helens reference DSM



MATHEMATICAL NOTATION

The following table sets up the mathematical notation used later. All homogeneous coordinates, including the scale factor ω (or 1), allow sum in matrix products. Components are equal to Euclidean space when ω equals 1.

Table 4: Mathematical notation in-use

MATHEMATICAL NOTATION IN-USE

Notations	Object side convention	Image side convention
Coordinates components	$\begin{cases}\lambda;\varphi;H \in geographic \\ X;Y;Z \in ECEF\end{cases}$	$x;y$
Normalised components	$\widehat{\lambda};\widehat{\varphi};\widehat{H}$	$\widehat{x};\widehat{y}$
Vector in euclidean space	$\boldsymbol{X}=\begin{pmatrix}X\\Y\\Z\end{pmatrix}$	$\boldsymbol{x}=\begin{pmatrix}x\\y\end{pmatrix}$
Vector in homogeneous space	$\widetilde{\boldsymbol{X}}=\begin{pmatrix}X\\Y\\Z\\\omega\end{pmatrix}$	$\widetilde{\boldsymbol{x}}=\begin{pmatrix}x\\y\\\omega\end{pmatrix}$
RPC function and inverse RPC function	$RPC(\lambda,\varphi,H)$	$RPC^{-1}(x,y,H)$
Geographic to Cartesian conversion and reverse	$G2C(\lambda,\varphi,H)$ $C2G(X,Y,Z)$	
Matrix and inverse matrix	$\boldsymbol{Y};\boldsymbol{A};\boldsymbol{X}=[X_1 \quad X_2 \quad \ldots];\boldsymbol{A}^{-1}$	

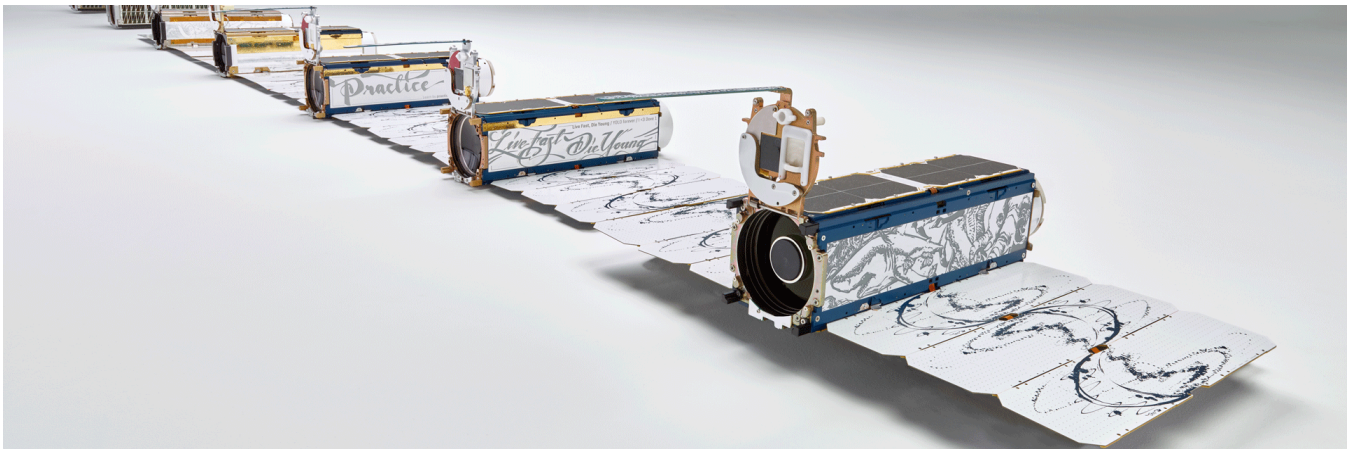
1. PLANETSCOPE

This part presents the Planetscope constellation in detail. All hardware and software descriptions are under a non-disclosure regulation. This brief outline brings some light on the current setting and forthcoming challenges.

1.1. FLEET

Planetscope is a satellite constellation made of approximately 130 CubeSat 3U divided into 12 sats flocks. Dove is the nickname of these satellites. They fly on a sun-synchronous orbit capturing frame-based images. The full fleet is able to cover the whole Earth's surface every day, and thus ensures a daily revisit period. It is the selling point claimed by the company.

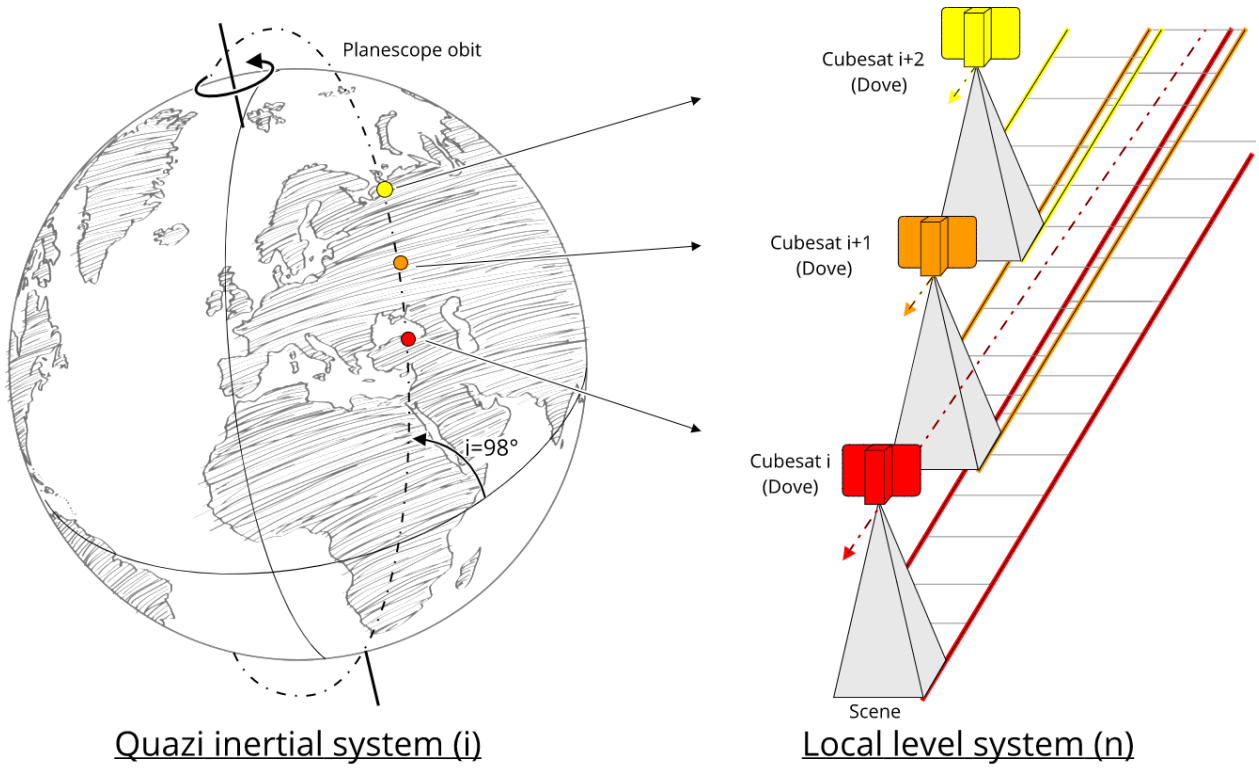
Figure 4: CubeSat 3U, so-called Dove



1.2. FLIGHT

All doves turn along decreasing orbits of height from 580 to 450 km. The orbit inclination is around 98°. These sun-synchronous orbits pass the Equator between 9:30 and 11:30 am (local solar time). The corresponding ground sampling distance (GSD) varys from 5 to 3 m and we would assume it at 4m. The frame footprint has approximately 24 km x 16 km or 32.5 km x 19.6 km according to the satellite generation. As displayed in the sketch below, all satellites per flock follow each other and capture the next ground strip as the Earth rotates. satellites fly *"downward"* during bright local time, an acquisition mode named descending orbit. On the other side of the orbit, satellites fly *"upward"* named ascending orbits. All doves flocks are set on 2 sun-synchronous orbits with 98° and -98° inclinaison angles ensuring whole Earth coverage. Hence, ground points may be sensed in ascending and/or descending orbit. Except the sensing time, there are no major differences between both orbit types since satellites point nadir.

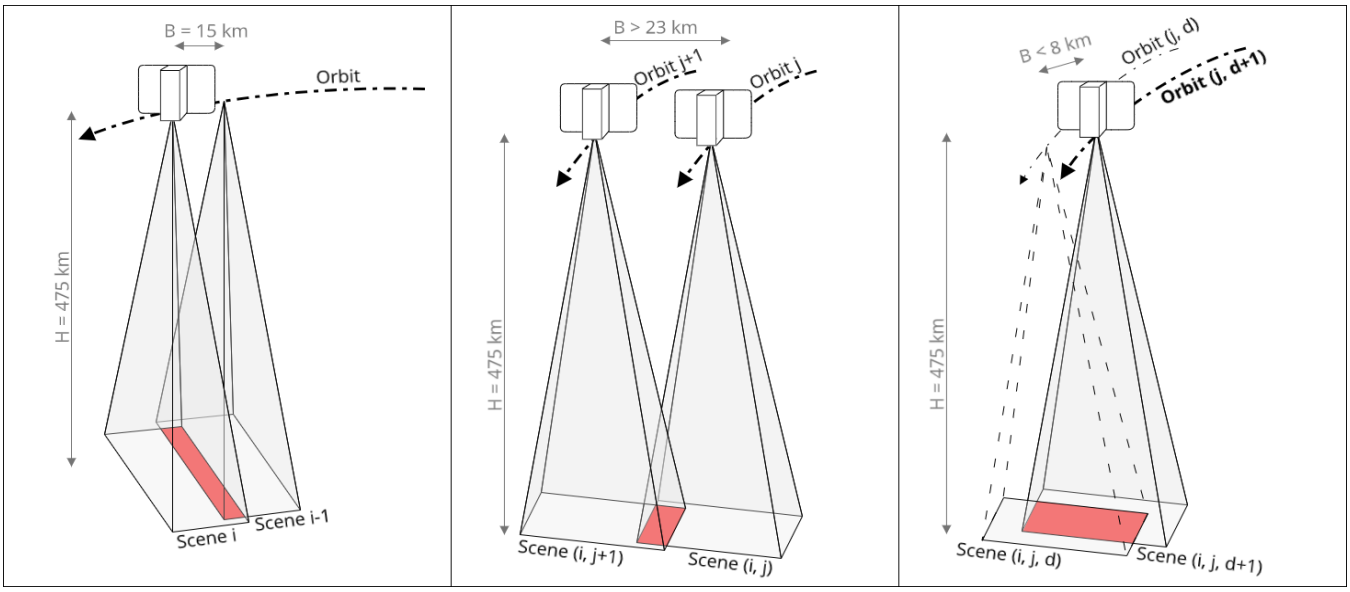
Figure 5: Dove flight along 1 orbit, left: Celestial Reference System (CRS), right: Earth fixed system



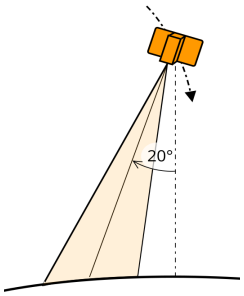
Since orbit heights decrease, satellites slightly accelerate during their whole life. It changes the satellite's nominal time at the equator crossing making it earlier. Hence, youth satellites pass the equator around 11 am in local sun time and old satellites pass earlier in the morning. Hence, It is common to find dark images sensed by old satellites with a low sun elevation, especially in winter. These dark scenes may be unusable because the whole image histogram is encoded with only a few bits close to zeros and it leads to quantization simplification.

We can classify overlapping areas into 3 classes: in-track, cross-track (cross strip) and cross date. The in-track overlap depends on several parameters computed aboard, ensuring at least 5% overlap to avoid any coverage gap. The cross-track overlap depends on satellite generations and satellite orbits in the quasi inertial system. Cross date overlaps happen because satellite orbits lead to different ground routes (local level system) from day to day.

Figure 6: Dove overlaps, left: in-track, middle: cross track, right: cross date



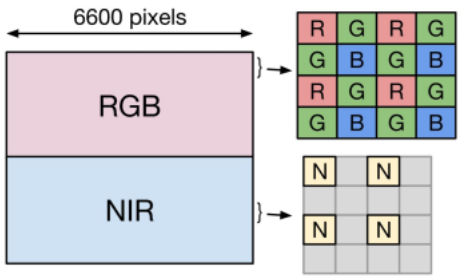
Like other Earth observation satellites, all Doves include several reaction onboard wheels. It is an orientation device which rotates the satellite body by braking or accelerating a constantly spinning mass. In normal operation, which is a nadir view, it adjusts the camera direction. During the internship, we scheduled several acquisitions with off-nadir angles. The reaction wheels rotate the satellite while it passes by the pole in order to cover a strip on the East or West side of the track. By doing so, we created several datasets of different off-nadir angles in the Planetscope database. These angles are between -20° (Eastward) and 20° (Westward). Those oblique acquisitions could be used in stronger photogrammetric blocks to improve the incidence angle at the ground. However, the limited time of the master thesis does not leave us the time to investigate its enhancement.



1.3. HARDWARE GENERATION

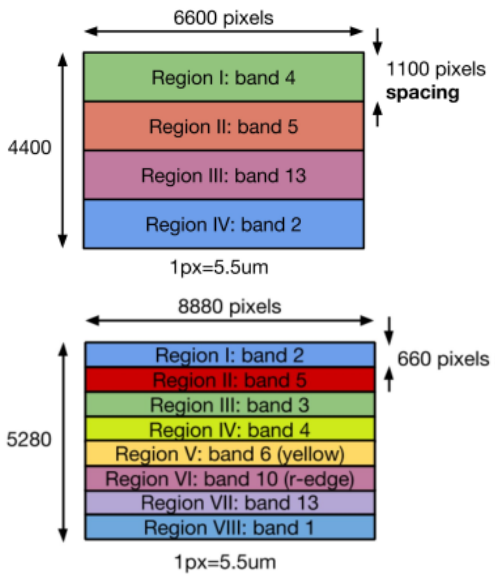
All these Doves are not identical to each other and 3 different satellite generations are in operation. The sensor design changed between generations with different hardwares.

After the Dove Pilot program, the first stable Dove generation is still operational: **Dove-Classic** (PS2). The hardware is a 6600x4400 pixel chip and a PS2 telescope. A Bayer pattern filter covers the full chip and an additional near-infrared (NIR) filter overlaps the bottom half. This chip division into sub-frames is called push-frame and a frame would appoint the whole chip size. The push-frame concept is extended in the following generations. Aboard a Dove-Classic, there is a PS2 telescope whose focal length is around 700mm.



Starting in November 2018, a new generation was launched: **Dove-R** (PS2.SD). It is based on the same hardware (chip and telescope) with a different filtering pattern. It is a full push-frame chip with 4 sub-frame capturing red, green, blue and NIR parts of a final scene and thus it avoids Bayer interpolation.

Finally, the 3rd generation came with more bands: **Super Dove** (PSB.SD). A new telescope PSB is mounted coupled to a larger chip (8880x5280 pixel). Also using a push-frame design, sub-frames are narrower than previous generations and it covers 8 different channels in the spectrum (coastal blue, blue, 2 greens, yellow, red, red-edge and NIR). The new telescope is similar to the previous one with 1mm longer focal length.



The sensor calibration in Planet pipeline is discussed in section 1.7. Another important point needs to be clarified about filters. Regardless of the Bayer filter, the sub-frame filter is an upper glass layer above the chip and its alignment is not perfect making the sub-frame divisions neither clearly limited nor identical on all satellites. We recognize in these 2 remarks the chain manufacturing method chosen by Planet which may introduce differences from satellite to satellite.

Both chip generations are divided into taps. That chip subdivision speeds up the reading time but may lead to some radiometric differences from tap to tap. The tap pattern depends on chip hardware (2x4 taps for 6600x4400 chips, 2x8 for 8880x5280 chips).

1.4. RADIOMETRY

Scenes are collected and stored in 12 bits. The ground processing applies a dynamic range scaling into 16 bits. This scaling transforms the relative digital number (DN) to absolute radiance values. In order to limit any image processing, we choose to work with the original DN. The 12 bits are encoded at 16 bit depth. All Structure from Motion (SfM) processes shall run on panchromatic images and a grey image creation will be presented later.

Many radiometric response assessments were conducted, especially prior Super Dove design. We should not need to care about that component for photogrammetric processes but we keep it in mind for Dove-R and Super Dove feature matching.

1.5. IMAGE PRE-PROCESSING

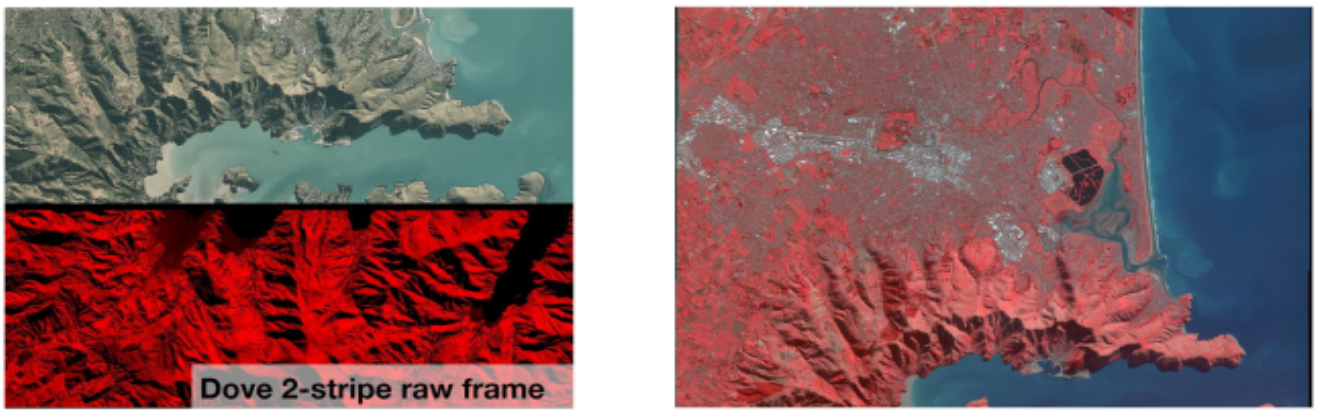
An internal process converts raw scenes (L0) to customer products (scene L1B, ortho-scene L3A, ortho-tiles L3B). The process workflow depends on the satellite generation, even though we can say that L1A step, which is not a customer product, includes radiometric correction from sensor defects and L1B involves geometric adjustment and projection. Every modification beyond L1B does not interest us because our photogrammetric computation needs parallax. As we also require the ”rawest” dataset, we conclude that the L1A product shall be the best product because it already includes corrections from sensor radiometric defects and it does not include geometric modification. Moreover, L0 scenes are restricted products that Planet cannot release while L1A scenes are less regulated even though they are not widely released.

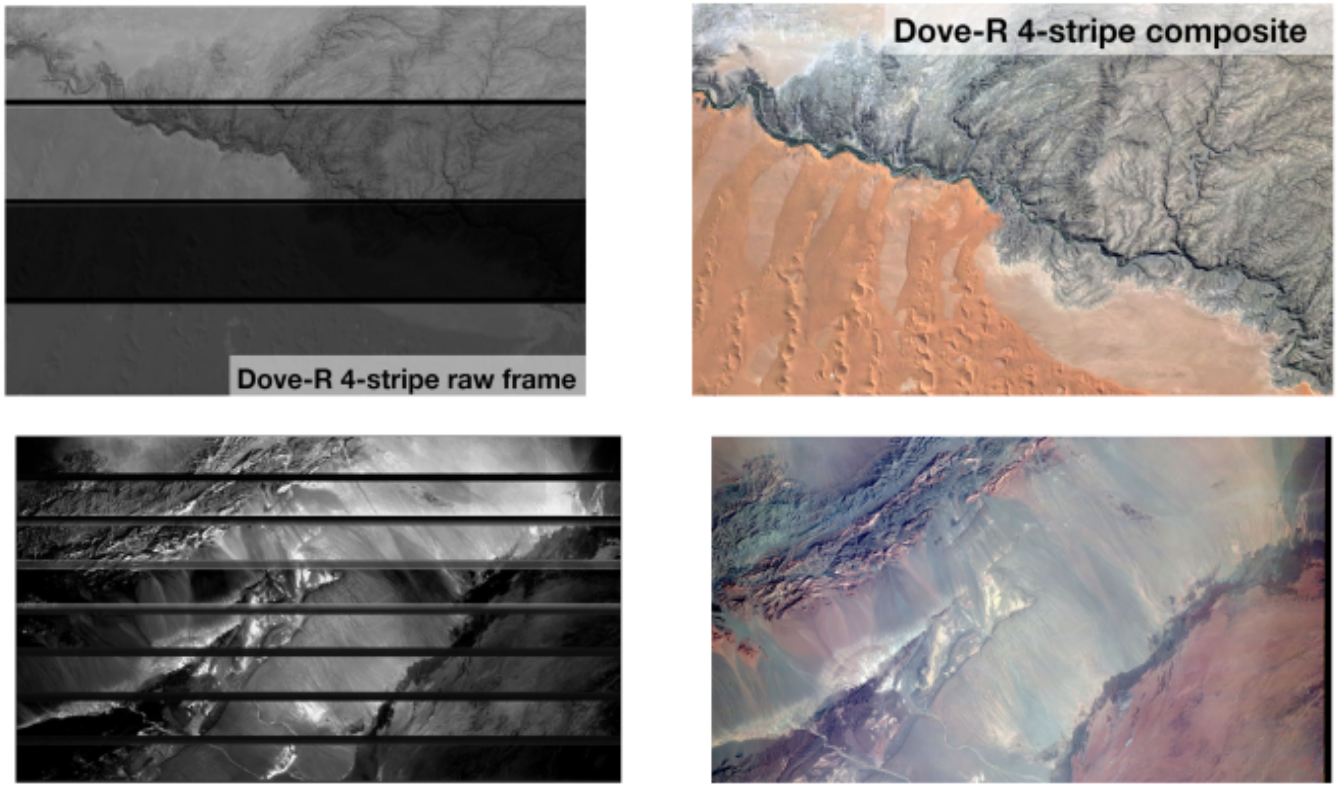
L1A scenes do not include any geometric modification. The main corrections applied to L1A are flat field and dark field. Flat field is a reference image which is the average of thousands of images. We expect a homogeneous grey image (flat) but there are some internal defects that appear like tap divisions. A product between the inverse of the reference image and a new one corrects these defects. The dark field is a subtraction bringing dark pixels to black (DN=0). Moreover, Dove-Classic L1A scenes include the co-registration and re-projection of the NIR band from the previous and next in-track image. That geometric transformation affects our process and we rely only on RGB bands from the L1A products. Hence, L1A is only a camera model and radiometric correction, since we rely only on RGB bands of Dove-Classic.

The L1B step is similar to the NIR band of Dove-Classic in L1A. The main scene (anchor frame) is co-registered to following and previous in-track scenes. These intermediate frames provide the missing sub-frames of different bands to complete the full anchor frame in every band. That process is called scene to scene registration (S2S). The registration depends on tie points between images which cannot be guaranteed in case of cloudy images.

The following figures display L1A (left) and L1B (right) images of different generations, except for Dove-Classic which is a L0 scene (left).

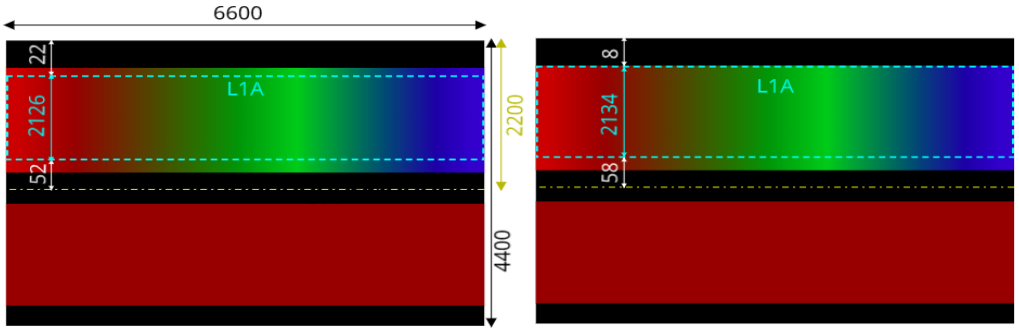
Figure 7: L1A and L1B products per generation: Dove-Classic, Dove-R, Super Dove





From the previous figure we observe that Dove-R or Super Dove L1A products keep the full image frame while a Dove-Classic L1A corresponds to the RGB sub-frame. Furthermore, the RGB sub-frame of a Dove-classic L1A had been cropped to avoid constant artefacts like TDI grey ramp over the first pixel rows or filter misalignment with the chip. The hereafter location models describe full size frames. In the following sketch of Dove-Classic, L0 frames (full size) have the sensor size and a L1A frame is the cropped RGB subframe (cyan). That crop deletes the TDI grey ramp at the image top and other edge effects at subframe borders. RPCs are valid over a 6600x4400 frame. However, within Dove-Classic cameras, there are several filter alignment and technologies and not all L0 scenes are cropped the same. The origin offset (top-left corner) is recorded in image metadata and included into downloaded RPCs. As a matter of fact, we are not concerned by origin offset in RPC applications but we extract the shift from them for physical model creation, as outlined in section 2.2.

Figure 8: Dove-Classic frame coordinates, left and right: 2 different crop types, cyan: L1A frame, white: number of cropped rows



1.6. RATIONAL POLYNOMIAL COEFFICIENTS

During the Planet pipeline, there is a geolocation process running in "parallel" to the S2S alignment. The initial information, provided by the operation team, describes the satellite orbit and its orientation. That team is responsible for satellite launch, status, monitoring, tasking (skysat) and other space activities. The orbit comes from a ranging method (telemetry) with one antenna in Norway for instance. That method provides accurate orbit elements. The orientation comes from star cameras matched to reference star maps. Unlike telemetry, retrieved orientations are not accurate. Positions expressed in Kepler elements are assumed to be physical models (satellite PM) because they present practical meanings. Nevertheless, their accuracy is about several kilometres and they must be adjusted.

Instead of projection matrices used with UAV of airborne acquisition, the satellite community uses Rational Polynomial Coefficients (RPC). We separate projection matrices under the name of physical models (PM) related to their physical meaning and RPC under the name of mathematical models. The section 2.2 details these descriptions and their differences. As a satellite image provider, Planet releases associated RPC models to its scenes. As presented in the annex 2, satellite PMs are transformed to RPCs at the pipeline beginning, named estimated RPC. Then, a layer-based adjustment refines initial models of anchor frames to "Seth RPC", using ground locks between the scene and a reference base-map coupled to a reference DEM. The process interpolates additional RPCs for non-anchor frames from the registered models but they are not used further. Finally, all models from the same strip build one solver that tweaks location model coefficients until the 1st order to final formula, named BA RPC. As described in the Planet documentation, that process chain ensures projection absolute accuracy (object to image) of 2.5 pixels (RMSE). It allows accurate orthophoto creation used for remote sensing studies but it does not fulfil a sub-pixel epipolar constraint used during computer vision processes. Hence, we include a refinement method in section 2.2.

However, the most accurate RPCs (BA RPC) are computed between L1A and L1B products for anchor frames. In the case of Dove-classic, all scenes are anchor frames and these RPCs are available for all images. In the case of Dove-R and Super Dove, only ⅛ or $^{1}/_{16}$ scenes are anchor frames coming with BA RPCs and others hold only interpolated RPCs. We would benefit from intermediate scenes for overlap coverage even though outcoming stereo geometries hold small incidence angles. Hence, the usage of intermediate Dove-R and Super Dove frames (not anchor frame) have to include the accuracy difference with respect to anchor frames during our refinement method. That study is not discussed in the following and it may be the next work step.

1.7. INTRINSIC CALIBRATION

As we saw before, the RPC creation requires physical information, extrinsic and intrinsic ones, that are merged into a mathematical expression. From a computer vision approach, the collinearity equation uses the interior geometry modelled as a pinhole camera, with focal length, principal point coordinates and pixel size, increased by a distortion model. After the object to image projection, that distortion description (model and coefficients) fits the pinhole image plane to the effective one.

There are several distortion models on a range of agility to cover deformations of the image plane and their calibration can be done over calibration field with targets (in-lab) or during the bundle adjustment, so-called in-situ modelling. However, the strong correlation between intrinsic and extrinsic parameters makes in-situ calibration complicated. That is why airborne camera manufacturers provide accurate models, in-lab computed, which are often recalibrated before purchase and even slightly corrected during following projects. Fraser *et al.* (1997) publication aimed at a stable method for intrinsic refinement during the bundle adjustment in the case of close range photogrammetry.

Among several, we have a closer look at the Brown-Conrady and the Tsai distortion standards, printed side to side hereafter. As released by Brown *et al.* (1966), completed by De Villier *et al.* (2008), the Brown-Conrady formula (left) describes the distortion removal, with radial component ($\mathbf{\Delta x_r}$), decentering component ($\mathbf{\Delta x_d}$) and distortion centre offset ($\mathbf{x_c}$). Conversely, OpenCV implements a distorting formula from Tsai *et al.* (1987) work (right), applying reverse correct in projected space (unitless) also called normalised coordinates (**x'**). From an image plane, undistorted points have to be projected to camera space before distortion application, and then reprojected to image space. There is no distortion offset included into that model and the decentering effects are only covered by $\mathbf{\Delta x'_d}$ coefficients. If we use the identity matrix instead of camera matrices, we run in pixel units and sole direction matters.

Equation 1: Brown-Conrady model

$$\begin{pmatrix} x \\ y \end{pmatrix}_{undistorded} = \begin{pmatrix} x \\ y \end{pmatrix}_{distorded} + \boldsymbol{\Delta x}_r + \boldsymbol{\Delta x}_d$$

$$r = \|\boldsymbol{x}_d - \boldsymbol{x}_c\|$$

$$\boldsymbol{\Delta x}_r = \left(K_1 r^2 + K_2 r^4 + K_3 r^6\right).\left(\boldsymbol{x}_d - \boldsymbol{x}_c\right)$$

$$\boldsymbol{\Delta x}_d = \begin{bmatrix} 2.P_1.(x_d - x_c)(y_d - y_c) + P_2\left(r^2 + 2(x_d - x_c)^2\right) \\ P_1\left(r^2 + 2(y_d - y_c)^2\right) + 2.P_2.(x_d - x_c)(y_d - y_c) \end{bmatrix}$$

Equation 2: OpenCV formula

$$\widetilde{\boldsymbol{x'}} = \frac{\boldsymbol{x} - \boldsymbol{x_c}}{\boldsymbol{f}} = \boldsymbol{K}^{-1}.\widetilde{\boldsymbol{x}}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix}_{distorded} = \begin{pmatrix} x' \\ y' \end{pmatrix}_{undistorded} + \boldsymbol{\Delta x'}_r + \boldsymbol{\Delta x'}_d$$

$$\boldsymbol{\Delta x'}_r = \boldsymbol{x'}_u\left(K_1 r'^2 + K_2 r'^4 + K_3 r'^6\right) \; with \; r' = \|\boldsymbol{x'}_u\|$$

$$\boldsymbol{\Delta x'}_d = \begin{bmatrix} 2.P_1.x'_u.y'_u + P_2.\left(r'^2 + 2x'^2_u\right) \\ P_1.\left(r'^2 + 2y'^2_u\right) + 2.P_2.x'_u.y'_u \end{bmatrix}$$

In the case of Planetscope, we have seen 2 different telescopes mounted on the 3 Dove generations: PS2 and PSBlue. Furthermore, there are 130 Cubesats (Doves) which are different hardware units, all including different intrinsic parameters each. That makes each satellite unique with its own telescope type and individual specialisations. As a whole, both telescopes follow a pincushion distortion (distorted radius is longer than a pinhole model) which reaches around 25 pixels at image corners of PS2 and around 140 pixels for PSBlue. Such values have to be modelled by sufficient distortion formulas and coefficients, unlike Skysat cameras from Bhushan (2021) study which uses a 3.6 m focal length. The distortion report, written by Mr Martos as computer vision scientist at Planet, focuses on distortion since the Cubesat manufacturer ensures accurate focal length and principal point per unit. It records rather distortion coefficients according to the Tsai standard. These values

are stored in a configuration file per satellite hardware, like the instance below. They use Tsai formula but in image space with coefficients in pixel units. However, OpenCV does not provide reverse formulas (remove distortion) and the operation has been approximated by different coefficients in the same formula that we call complementary coefficients. That file also provides absolute coordinates of the distortion centre since `CAMERA_CENTER_X` and `_Y` are principal point coordinates. Even though these models use OpenCV convention (Tsai), its integration into RPC passes by hand written function.

```
'0e19':{'format': FORMAT_OPENCV,
        'chroma': CHROMA_MONO,
        'add':{'k1': -3.94502713e-10, 'k2': 0.0},
        'rem':{'k1':  4.00255009e-10, 'k2': 0.0},
        'center':{'x': CAMERA_CENTER_X - (11.02295 ),
                  'y': CAMERA_CENTER_Y - (169.37586)} }),
```

All coefficients above were not calibrated through the same way. The manufacturer provided an in-lab calibration of both telescopes with distortion formula and coefficient. Then, Planet scientists extracted new coefficients from that dataset and concluded that K_1 was sufficient to model the distortion of PS2 in both directions and K_2 was set to 0. For PSBlue, both K_s are required to model the large view angle in both directions. In both cases, the decentering parameters (P_1, P_2) are skipped since offset parameters (x_c, y_c) were included. Therefore, these coefficients were fixed for all hardware and centre offsets computed per hardware unit and fixed before all following processes. Then, launched satellites into orbit endured thermal and mechanical stresses on the telescope which changed physical distortion. In order to circumvent that issue, Planet scientists designed an in-situ calibration method from a large stack of scenes. It contributes to both telescope distortion with offset correction and it came up with K_s parameters of PSBlue for both applications (add and remove distortion). Even though it includes a very large number of scenes (several thousands) over known calibration sites like Batou in Mongolia, an in-situ calibration is correlated to other adjustment parameters (RPCs) and, in that case, it only fits rectification needs. Regardless of their physical meaning compared to in-lab calibration, they are the ones used for RPC creation. However, that distortion calibration refines distortion coefficients on a regular basis from an increased stack of scenes. It changes distortion coefficients from time to time that are included in extended metadata. New acquired scenes convert current distortion coefficients into RPC and the later distortion models do not update that location model. Hence, the usage of current distortion models on old scenes may cause discrepancies in RPCs. Finally, any remaining miscalibration is indirectly refined in RPC adjustment due their overparameterization. All in all, there is stable and meaningful available information but it does not take care of launch changes and there are also correlated and evolving in-situ parameters. In section 2.3, we make our parameter selection and refine some of them.

2. DEVELOPMENT AND IMPLEMENTATION

That development step gathers the assessment of former studies, adds some parts and designs the process chain using python and ASP library. As lately formalised by Schonberger (2016), a standard SfM workflow, is divided into 3 main parts: correspondance search, incremental reconstruction, and dense matching. Starting from that, our full chain adds a block creation step before, named Stereo-Scene Block Parsing (SSBP): section 2.1. It merges the correspondence search and the incremental reconstruction into one step called Absolute Structure from Motion (ASfM): section 2.2, and it keeps the dense matching part called Multi-Scene Stereo (MSS): section 2.3. That development hereafter runs over Providence mountain AOI.

2.1. STEREO-SCENE BLOCK PARSING (SSBP)

2.1.1. PURPOSE

All photogrammetric computation requires accurate relative positioning ensured by the bundle adjustment, and thus that first step aims to extract the best scene selection from the whole database. We have seen that PS flies for more than 6 years and the data collection increases everyday. Nowadays, the entire storage contains several Terabytes of scenes. The SSBP step aims to parse along the whole scene database and select a relevant block of scenes. Planet's IT organisation is part of the study but it is not presented here. Then we run an efficient clustering tool, named SSBP, which reduces the number of scenes by selection of the best ones.

An additional sub-brick supplements SSBP, called Planet Common Tool (PCT). It is an interface for product creation through the Planet's IT setting. We have seen that L1A scenes are the required products but these images are not stored. PCT creates requested products in the Planet system and downloads it. Its coarse design is presented after since it remains a straightforward implementation.

2.1.2. ALGORITHM DESIGN

There is an infinite number of scene combinations over a single point but the user shall have some strict requirements like the area and the acquisition period. Therefore, those criteria provide the starting point as a web query uploaded to Planet customer API. In addition to the AOI shape and the acquisition period, it carries a cloud percentage filter, a Dove generation filter, a product type filter, etc. That query is assessed by Planet API running on Google Cloud Platform (GgCP) and it is able to go efficiently through the database and return a json file with scene descriptions. That file contains standard metadata of matching scenes with a referencing ID, footprint geometry, and other parameters. This descriptor outlines our strictly filtered selection.

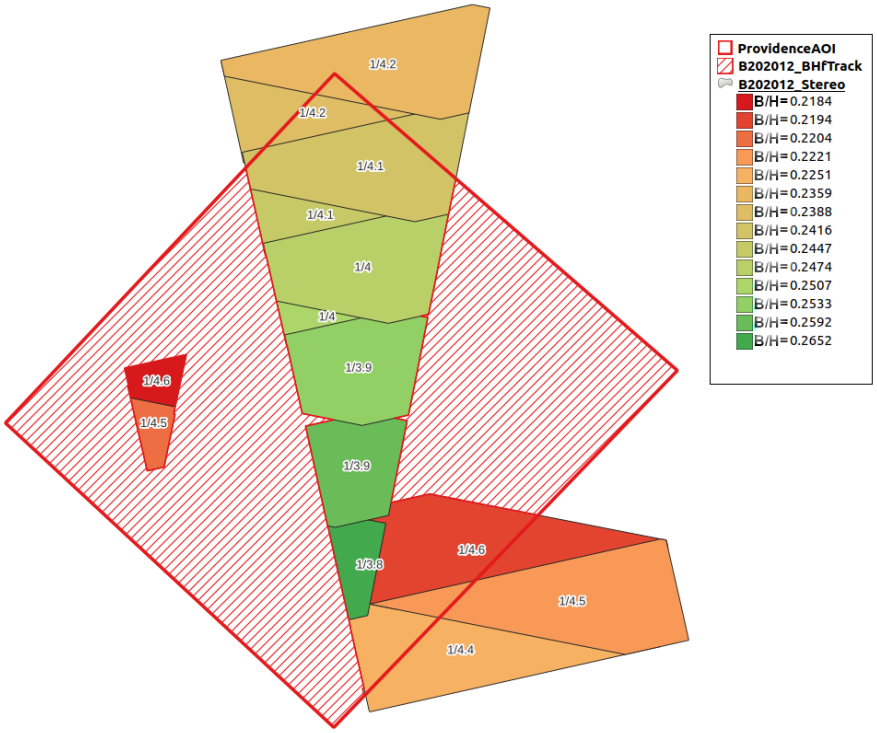
In order to make large queries available, the next step divides the strictly filtered list into scene blocks. A basic option considers the whole list as one block but a *"month"* option creates blocks based on the acquisition date, all scenes of a month gathered into a block. This simple clustering

method uses standard metadata and writes the block descriptors in json format. We have not implemented other clustering methods because these ones seemed enough for development usage.

Even after month clustering, there are a large number of scenes per block, especially over dry regions. In optimal cases, each scene footprint provides 30 images due to PS time resolution. Another filtering came up using these footprints written in standard metadata. Two scenes are considered similar if more than 80% of their footprint overlaps. If they overlap more than the threshold and they have similar satellite azimuts, it means that camera centres are close and the function attempts to extract the best image using cloud percentage, sun elevation angle and quality tags. In case of equality, it selects the first image. Hence the number of scenes decreases again by skipping identical view perspectives and it updates descriptors with the filtered selection.

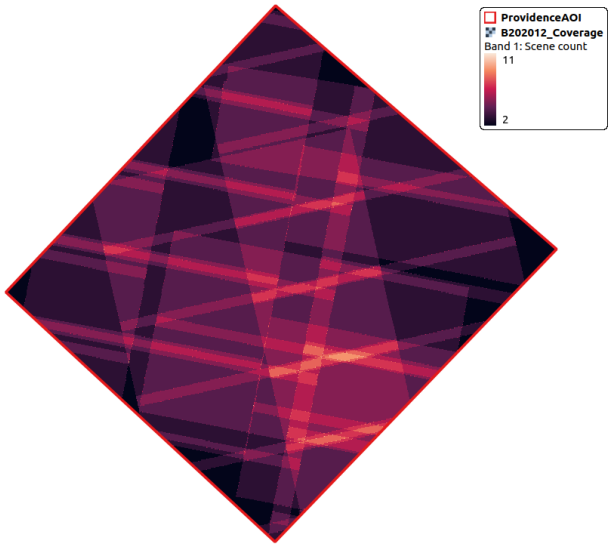
Finally, we implement the last filtering method maximising the B/H ratio. In photogrammetry, the Base-Height ratio holds a direct relation to the incidence angle and the match accuracy from the current stereo pair. A large B/H ratio corresponds to a large view angle difference and an accurate intersection while a small value corresponds to narrow incidence angle. Aati (2020) used the same criterion to distinguish scenes. The filtering is implemented with geometric operations remaining fast. Computed by the Operation team quoted in section 1.6, the initial camera position is read from extended metadata and the B/H ratio is set to the footprint intersection geometry. Then, along a sorted list, with respect to B/H ratio, we fill up the AOI shape. New couple geometry intersecting the AOI polygon is stored with its scene ID and removed from the AOI shape. Once the AOI shape disappears, the entire scene list provides the best scene couple. By default, selected scenes assume other stereo pairs called default redundancy. Although, a redundancy parameter improves that function by forcing the same computation with multiple AOI polygons. Hence, a B/H filtering with redundancy 2 fills 3 AOI polygons and returns 3 best stereo pairs over each point that is boosted by the default redundancy. That filtering method is light with geometric handling but it requires a stereo pair description. Therefore, we apply a footprint filter beforehand limiting the number of scene combinaisons. Nevertheless, the initial position in extended metadata is not accurate and there is a B/H ratio difference to the truth. That filter stores a backtracking file with evolving AOI polygons. It is presented in the following figure at an intermediate stage (hased red) stereo pair footprints (solid colour); the colour rampe follows the decreasing B/H order.

Figure 9: BH filtering, red: AOI geometry, hased red: filter backtracking file, solid colour rampe: stereo footprints



Beside the json descriptor, SSBP stores other depiction files. It converts the json descriptor to a geojson file for GIS software as well as stereo pair polygons (displayed above). Furthermore, the scene list is stacked into a raster file holding the number of scenes per pixel, as displayed hereafter. All these descriptors are useful user interfaces.

Figure 10: Block coverage, raster descriptor



After clustering and filtering parts, descriptor files describe block contents. These scenes exist on the Planet cloud as L0 and L1B products. The intermediate level L1A was not stored and the Planet Common Tool sub-part (PCT) holds Planet settings for image creation through cloud process. Planet stores and processes databases on Google Cloud Platform (GgCP) that we task for product creation. As displayed hereafter, PCT links block descriptors, cloud buckets and local storage with the *"info"* option. The cloud bucket name (e.g. valintern_dsmps_B202012_L1A) is hard coded in PCT script and scenes are called *"Features"* as json heritage. The *"match"* function compares those 3 space contents, the *"create"* option tasks the image creation and the *"download"* option downloads them to the local storage.

Table 5: Product creation status, left: descriptors, middle: GgCP bucket, right: computer storage

```
[11/16 12:57:21 pct_main]: B202012
[11/16 12:57:27 pct_main]: valintern_dsmps_B202012_L1A (exist: True)
       |   Descr  |   Cloud  |   Local  |
Nb Feat|    42    |   152    |    42    |
Size   |    /     |  8.7 GB  |  2.3 GB  |

[11/16 12:57:27 pct_main]: B202101
[11/16 12:57:31 pct_main]: valintern_dsmps_B202101_L1A (exist: True)
       |   Descr  |   Cloud  |   Local  |
Nb Feat|    38    |   102    |    38    |
Size   |    /     |  6.0 GB  |  2.2 GB  |
```

After the scene selection, and their creation, the local system stores a block repository with block descriptors, L1A scenes, highest RPC files, all ready for following photogrammetric steps. In relation to product creation, the extraction of panchromatic images is presented here, even though it runs during ASfM. Computer vision requires single band scenes. In the case of Dove-Classic, L1A products are 4 bands, but only RGB channels are unaltered. We favour the extraction of green band over HLS (or HSV) colour space change because green band standard deviation remains larger (or same) than lightness channel. In the case of Dove-R and Super Dove, L1A products are single band images encoding different spectrum channels, the so-called push-frame system. Hence, there is no need for single band extraction but rather mask management in feature extraction. That extension is not discussed here.

2.2. ABSOLUTE STRUCTURE FROM MOTION (ASfM)

2.2.1. PURPOSE

A major step of the Structure from Motion (SfM) chain is the refinement of location models that Schonberger (2016) named incremental reconstruction. His work started with unknown physical models while PS scenes come with Rational Polynomial Coefficient models (RPC), outlined in the section 1.6 and we would prefer the bundle block adjustment naming. These location models ensure a 10 m (2.5 pixel) projection, as written in the Planet documentation, which is a sufficient constraint for object to image projection with remote sensing application, but computer vision correlation requires sub-pixel accuracies. We confirmed it with relative comparison of orthophotos. The epipolar constraint would not be valid without bundle adjustment refinement. Unlike Schonberger (2016), we do not start the reconstruction from scratch and we refine the location model during the Absolute

Structure from Motion part (ASfM). It runs a correspondence search and a bundle adjustment that we attempt to summarise into a single value as block relevance. It is mainly based on the Ames Stereo Pipeline (ASP) library in order to make processes efficient and stable. Therefore, it involves a good understanding of the software. Once final residuals are accepted, final models become input data of the next step, MSS.

The *"absolute"* component in the title indicates that we limit the use of external data sources as possible. All available information about Planetscope shall hold enough parameters and we expect no need to add others. In terms of coordinate systems, the *"absolute"* concept is also related to the ECEF system used by ASP instead of geographic coordinates, relative to ellipsoid, or local projections. We also consider camera models in millimetres instead of pixels in order to avoid pixel size dependency.

As we conclude in section 1.7, that step also aims to improve intrinsic parameters of each satellite. We assume cameras to be stable in space and thus intrinsic parameters are corrected per satellite unit.

2.2.2. ALGORITHM DESIGN

The satellite community used to work with Rational Polynomial Coefficients (RPCs) instead of physical models (PMs). The RPC is a standardised and flexible mathematical tool to fit pixel-wise projection. Its authenticity was proved in Fraser *et al.* (2003) publication with one coefficient fixed to 1. Zhang *et al.* (2006) released a mathematical description with geographic coordinates. As shown after, they transform geographic coordinates to pixel coordinates in one step, the so-called object to image transformation. It is achieved with a ratio of 3rd order polynomials with 40 coefficients in total for each image component (row/column or line/sample) which had been normalised by offset and scale coefficients (not written after) in order to preserve the computation accuracy. Beside, we write the physical model standard (PM) which also applies an object to image projection by collinearity equation. It is the historical basic formula and it remains the favoured method by the airborne community because it applies meaningful coefficients (focal length, position, etc). PM also includes a distortion model (formula and coefficients), not written below. Satellite providers prefer to hide physical properties and they came to the RPC models. They transform orbit Kepler elements, star camera orientation and calibration parameters to RPC before release. A reverse conversion is not exactly possible and many publications speak of it. In addition, RPC coefficients mix several effects and the final model leads to overparameterization. All transformations, ellipsoid to cartesian, cartesian to camera, camera to pixel coordinates, may be approximated by individual polynomials but their heap requires flexible models with extra parameters. As we saw in section 1.6, RPCs are created at an early stage of the Planet pipeline which adjusts their coefficients and they are our input data holding the Planet accuracy. Both models are side to side hereafter.

Equation 3: Mathematical model, Rational Polynomial Coefficient (RPC)

$$\widehat{x} = \frac{P_x(\widehat{\lambda},\widehat{\varphi},\widehat{H})}{Q_x(\widehat{\lambda},\widehat{\varphi},\widehat{H})} \text{ and } \widehat{y} = \frac{P_y(\widehat{\lambda},\widehat{\varphi},\widehat{H})}{Q_y(\widehat{\lambda},\widehat{\varphi},\widehat{H})}$$

$$\text{with } P,Q_{x,y} = \sum_{i=0}^{3} \sum_{j=0}^{i} \sum_{k=0}^{j} a_m \widehat{\lambda}^{i-j} \widehat{\varphi}^{j-k} \widehat{H}^{k}$$

$$\text{and } m = \frac{i(i+1)(i+2)}{6} + \frac{j(j+1)}{2} + k$$

Hence,

$$\hat{x} = \frac{a_0 + a_1\hat{\lambda} + a_2\hat{\varphi} + a_3\hat{H} + a_4\hat{\lambda}\hat{\varphi} + a_5\hat{\lambda}\hat{H} + \dots}{1 + b_1\hat{\lambda} + b_2\hat{\varphi} + b_3H + b_4\hat{\lambda}\hat{\varphi} + b_5\hat{\lambda}\hat{H} + \dots}$$

$$\hat{y} = \frac{c_0 + c_1\hat{\lambda} + c_2\hat{\varphi} + c_3\hat{H} + c_4\hat{\lambda}\hat{\varphi} + c_5\hat{\lambda}\hat{H} + \dots}{1 + d_1\hat{\lambda} + d_2\hat{\varphi} + d_3H + d_4\hat{\lambda}\hat{\varphi} + d_5\hat{\lambda}\hat{H} + \dots}$$

Equation 4: Physical model, Projection Matrix (PM)

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = G2C(\lambda, \varphi, H) = \begin{pmatrix} (n+H)\cos\varphi\cos\lambda \\ (n+H)\cos\varphi\sin\lambda \\ ((1-e^2)n+H)\sin\varphi \end{pmatrix}$$

$$\widetilde{\boldsymbol{x}} = \boldsymbol{P}.\widetilde{\boldsymbol{X}} = \boldsymbol{P}_{3\times4}.\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$\boldsymbol{P}_{3\times4} = \boldsymbol{K}_{3\times3}.\boldsymbol{O}_{3\times4}$$

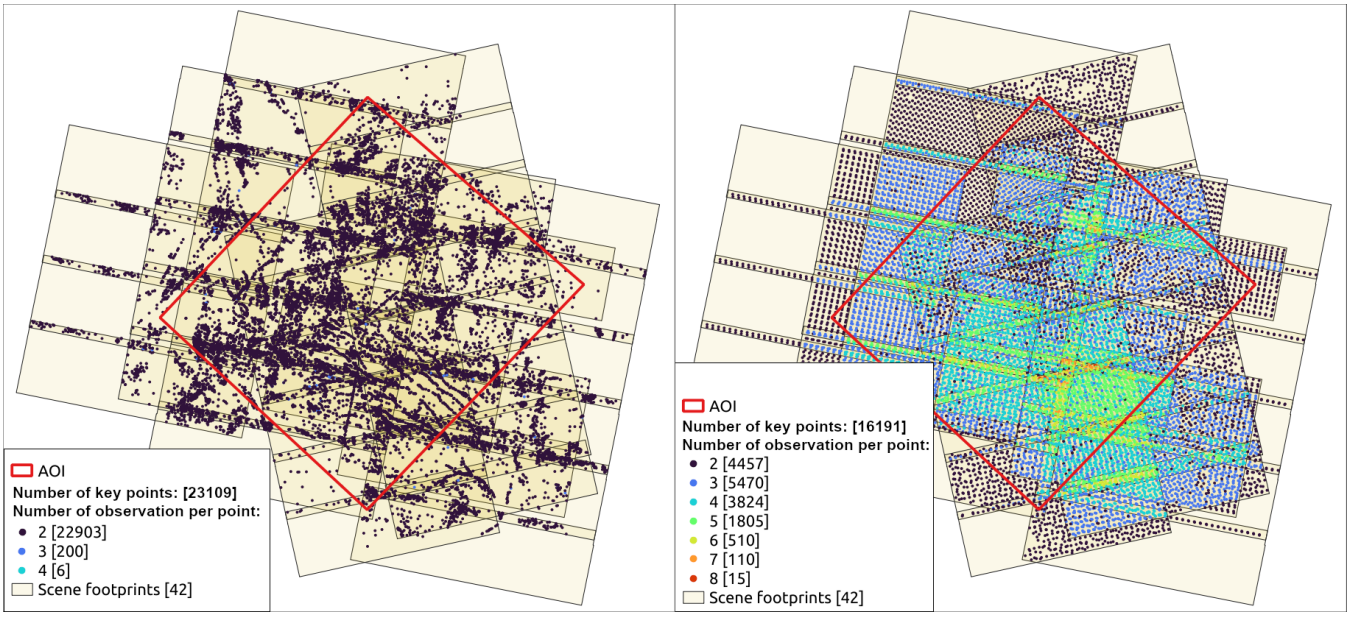
$$\boldsymbol{O}_{3\times4} = [\boldsymbol{R} \mid -\boldsymbol{R}\boldsymbol{X}_0]$$

$$\boldsymbol{K}_{3\times3} = \begin{bmatrix} m_x f & 0 & P_{0,x} \\ 0 & m_y f & P_{0,y} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & \rho \end{bmatrix}.\rho^{-1}$$

Many softwares are able to read RPCs like GDAL but few of them provide a stable solution to adjust them. The object to image function is straightforward making the orthorectification easy but there is no adjustment convention. In the SfM pipeline formalised by Schonberger (2016), the correspondence search makes no use of location models and includes only image processing with SIFT extraction, feature matching and RANSAC filtering. It only considers location models (if existing) during the incremental reconstruction triangulating features into key points. This reconstruction requires a batch of faithful features well distributed. In addition to RANSAC iteration ensuring feature quality, we include a triangulation filtering and fix an epipolar threshold with respect to the RPC accuracy. The software supports RPCs during feature extraction and thus our epipolar filtering is fixed at 2 pixels. In ASP, the correspondence search runs at the beginning of *bundle_adjust* and *stereo* functions. The *stereo* function is a packed workflow which computes image transformation (affine or homography) of image pairs based on extracted features before dense matching. The workflow description is presented in annex 3. For unknown implementation reasons, *stereo* returns better features than the *bundle_adjust* function. We implement the feature extraction along a softness range. Starting from strict parameters like small uniqueness and triangulation filtering with small epipolar distance, the algorithm eases them whether too few points come out. It reduces the number of wrong matches and speeds up the feature extraction. Moreover, we guarantee the features distribution by an extended operation. After correspondence search, we compute image transformation and dense matching. Then, we use location models (RPCs), 2D transformation (affine or homography) and disparity map (from correlation) to predict features in the corresponding image. Finally, we run a feature extraction (image process) around the predicted point to retrieve accurate similar points. That image operation unlink feature coordinates to the whole chain. It runs over a grid providing almost regular key points. That process runs pairwise with RPCs models and returns spread and faithful features.

A fixed bundle block adjustment (fixed location model) with all pairwise features merges points into triplet, and even more observations. Multi-view features build a stiffer block geometry because they are harder constraints used by adjustments. The following figure displays a 42 scenes block and the feature distribution with observation number colour rampe. On the left is the sole image extraction and on the right is the extended grid creation. The grid option is a longer process but it returns a homogeneous distribution with more multi-view features. The overall number is lower for a better spread, reducing the computation effort during bundle adjustments.

Figure 11: ASP key point distribution and their observation number, left: image extraction, right: grid extraction



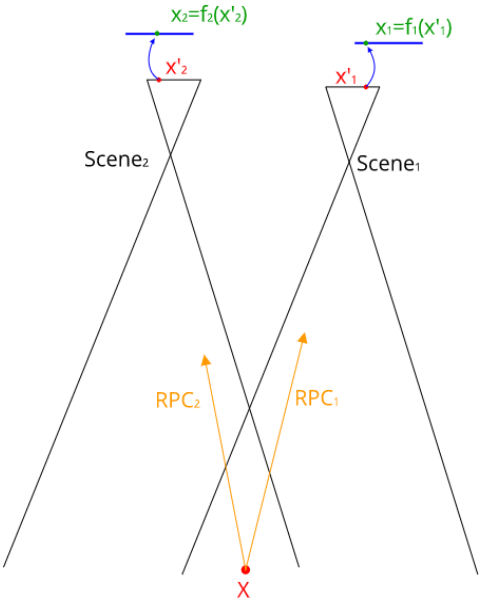
During that fixed adjustment, we also force key point heights to the SRTM in order to bring out the initial residuals of input RPCs. It makes use of the adjustment engine to project key point batches and resume their reprojection accuracy. Moreover, it lists key points (including their SRTM height) into a connectivity file that we use later. Hence, we observed that in-track correspondences hold sub-pixel accuracy but some cross track or cross date couples present 2 pixel residuals. It confirms the 2.5 pixels RPC accuracy and the significance of following bundle adjustments.

As we have seen before, there is no conventional way to adjust RPCs. For instance, Planet layer-based pipeline adjusts coefficients until the 1st order. On the other hand, Ghuffar (2018) and Aati (2020) computed a 2D transformation (often affine) per image plane, a method also implemented in the *stereo* function of ASP. That is a common way to adjust RPCs because it includes a stiff transformation in addition to the mathematical one and avoids overparameterization. That affine transformation mainly compensates extrinsic misalignments. The *bundle_adjust* function, in ASP, behaves differently and computes ground transformation per image with respect to a predicted image centre. Both methods are sketched hereafter. On the left, the function returns a 2D transformation, noted $\boldsymbol{A}$, which projects a virtual image plane returned by RPC to the real one. On the right, the function returns a translation and a rotation (quaternion) as additional transformations to RPCs, noted $\boldsymbol{G}$. It is important to notice that 3D transformation stands in ECEF coordinates while

the RPC requires geographic coordinates and equations below skip it. Like the 2D transformation, that object space transformation avoids overparameterization and compensates extrinsic misalignments. However, the usage of that transformation becomes tough out of ASP because it refers to an origin computed through a complex internal method as discussed on ASP GitHub. It defines a rough image centre from RPC parameters and point pair solver. Hence, the shifted transformation named $\widetilde{G}$ turns more accurate but its origin is not recorded.

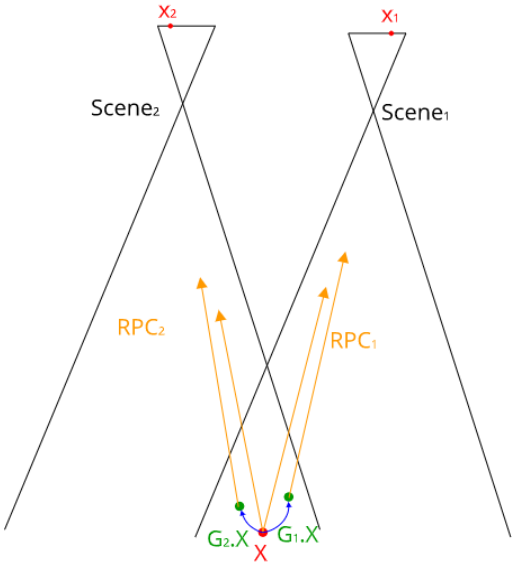
Equation 5: 2D transformation (ASP *stereo* function, Ghuffar (2018) and Aati (2020))

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} a_1 & a_2 & a_0 \\ b_1 & b_2 & b_0 \end{bmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathbf{A}. \begin{pmatrix} RPC_x(\lambda, \varphi, H) \\ RPC_y(\lambda, \varphi, H) \\ 1 \end{pmatrix}$$



Equation 6: 3D transformation (ASP *bundle_adjust* function)

$$\boldsymbol{x} = RPC_{x,y}(\widetilde{\boldsymbol{G}}.\boldsymbol{X}) \text{ with } \widetilde{\boldsymbol{G}} = \left[\widetilde{\boldsymbol{R}} \mid \widetilde{\boldsymbol{T}}\right]$$



Both methods are available in ASP but some issues remain with the RPC bundle adjustment. The 2D transformation implemented in *stereo* function does not support multiple images and runs pairwise. A proper method would have been a bundle adjustment type engine with A matrix solving features ground coordinates stepped up by 2D transformation coefficients along columns with respect to feature image coordinates along rows. That option was developed in COSI-Corr by Aati (2020) but we have not chosen to implement that other library in our system. If it were the chosen method, it would have been interesting to investigate the effect of homography or higher polynomial order transformation, correcting miscalibration and preserving the relative distribution from the existing dense feature grid. The 3D transformation implemented in the *bundle_adjust* function is a stiff and fast adjustment. However its origin is not written and is not well documented, as discussed on ASP GitHub. It is a camera centre approximation depending on the library setting. To use that result, we must merge the input RPC and the outcoming adjustment into a new RPC with the software itself but that function yields cropped scene RPCs (reduced factors). Last and not least, during 3D

modelling, we cannot create epipolar images with RPCs as a major aspect of our image based approach. ASP would retrieve an image transformation (affine, homography) computed on new features as explained before and block adjustment efforts would be changed to a local estimation. All these issues are related to mathematical model usage but a process based on physical models brings new perspectives. The adjustment standard with physical models is available in ASP and well implemented, leaving controle on each component. The epipolar transformation becomes possible before dense matching relying on globally refined models. With such an approach, the key step becomes the physical models creation before bundle adjustment.

A physical model (PM) is the object to image transformation by projection matrix. As explained before, that projection matrix packs the collinearity equation up to a single matrix product with homogeneous coordinates. However, the complete transformation, which was embedded into one mathematical model, has to be broken down to successive steps. It first converts geographic to cartesian ECEF coordinates, then points are projected to image frames with the projection matrix, and finally a distortion model adds camera characteristics. Projection parameters can be divided into extrinsic ones with the position and orientation of the camera and intrinsic ones with focal length, principal point coordinates, pixel size. There are large correlations between extrinsic and intrinsic parameters, like the direct link between focal length and distance to the object for instance. In ASP software, a PM is stored into a *tsai* file whose description is hereafter, and we convert it to a projection matrix before application as written beside. Following expressions skip distortion models.

Equation 7: *Tsai* file description and formulas

```
VERSION_X: standardisation version
PINHOLE: type of camera model
fu, fv: focal length (width, height)
cu, cv: principal point coordinates
(width, height)
u, v, w_diretcion: axis permutation
transformation
C: camera centre coordinates (ECEF)
R: rotation matrix (ECEF)
pitch: pixel size, it fixes the file unit
NULL|TSAI|Brown-Conrady|RPC: distortion
model with parameters after
```

$$\widetilde{X}_{ECEF} = R_i^o . \widetilde{x} + C$$

$$\widetilde{x} = R_i^{o-1} . \widetilde{X}_{ECEF} - R_i^{o-1} C$$

$$x = \frac{1}{p}\left(\frac{x_1}{x_3} f_u + c_u \, , \frac{x_2}{x_3} f_v + c_v\right)^T$$

Equation 8: *Tsai* file turned into projection matrix

$$\widetilde{x} = P . \widetilde{X}_{ECEF} \text{ with } P_{3\times4} = K_{3\times3} . O_{3\times4}$$

$$O_{3\times4} = \left[R_i^{oT} \mid -R_i^{oT} . C\right]$$

$$K_{3\times3} = \begin{bmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & pitch \end{bmatrix} . pitch^{-1}$$

The conversion from mathematical model (RPC) to physical model (PM) becomes the key step as we saw but there is no direct transformation. It would be possible to build a physical model from initial information (orbit Kepler elements, star camera attitude, intrinsic calibration). Although, we have seen that orientations are not accurate and we would lose benefits of RPC refinement in the Planet pipeline. Therefore, we prefer to approximate RPCs with point pairs (object and image spaces) and

fixed parameters. That computation is known as Spatial Resection (SRS) or Perspective n Point (PnP). There is a strong correlation between all parameters at that flight height and we need a large point grid to model extrinsic parameters solely and avoid Projection Direct Linear Transformation weakness (P-DLT). As detailed in annex 5, it starts with a point grid over the entire frame since RPCs define the L0 product. On one hand, we project them to several heights using inverse RPCs and, on the other hand, we correct their camera distortions as presented in section 1.7. We also fix the camera matrix with manufacturer information. Then, we fit these pairs by a PnP solver in OpenCV. The large height range of the grid combined with the Efficient PnP (EPnP) algorithm, from Lepetit *et al.* (2008), provide a stable method returning a sub-pixel approximation. Instead of this handmade function using OpenCV, it is possible to implement similar computation with ASP. It requires the computation of a no distortion RPC from corrected point pairs. Then, the *cam_gen* function projects an image grid to a given DEM and solves extrinsic parameters with given intrinsic parameters. The refinement option of that function corrects the coarse extrinsic parameters and overtakes the weakness of the first approximation. The inverse RPC computation is detailed in annex 4 and a comparison between both methods is presented in annex 5. As a whole, we prefer the OpenCV method because it runs without external DEM and the feature projection (during bundle adjustment) is closer to RPC projection.

We use distortion models presented in section 1.7 during PnP. Among the list of Browns-Conrady, Photometrix, distortion RPC, etc, we prefer to write down the Tsai formula because bundle adjustment can handle it and it remains the closest model to the Planet calibration. It was discussed by Tsai *et al.* (1987) and applies the direction undistorted-distorted like the Mr Martos estimation. However, it does not include distortion centre offset and uses normalised coordinate, so-called camera space. Therefore, the Martos' parameters (radial K_s, decentering P_s) must be converted to normalised coordinates and the decentering is set as principal point before PnP solvation, as described in annex 5.

The PM bundle adjustment is well-known and standardised. The solver adjusts features 3D coordinates as well as camera parameters. We let the engine run with camera PMs and tie points computed with RPC as explained before. In order to avoid adjustment drift, we include Ground Control Point (GCP) coming from RPC key points and SRTM height. We select them at position extremum (latitude, longitude, height) and observation extremum too, meaning 4 times 10% of the scene number in totals. Their ground standard deviation is fixed to 10 m (2.5 pixels) according to the RPC accuracy and the image standard deviation equals 1 pixel even though we expect sub-pixel extraction. Since we are not sure whether extrinsic-intrinsic correlation is properly managed by the ASP engine, we prefer to run a bundle adjustment in 2 consecutive phases: EO-BA refining only extrinsic parameters and then IO-BA including intrinsic parameters with larger weight on already adjusted extrinsic ones. Both steps make use of the same feature batch and same GCPs. Hence, EO-BA corrects PnP errors and camera misalignment at a time, and then IO-BA uses the accurate solution for intrinsic perfection.

As we saw in section 1.7, distortion coefficients are in-lab solutions with a limited freedom while distortion centre offsets are in-situ approximations. Furthermore, the centre of symmetry set as principal point in Tsai model affects the decentering modelling. Therefore, we prefer to adjust only principal point coordinates during IO-BA. By default, ASP creates adjustment bundles from identical

values and thus all scenes from the same satellite are adjusted by the same amount, fulfilling the refinement per satellite need. The next table shows the comparison of principal point coordinates before (from Planet distortion) and after adjustment where some are slightly corrected (*e.g.* sat 1048) and others end in a symmetric position (*e.g.* sat 103c) with respect to the frame centre. Hence, this adjustment is mandatory to circumvent incoming errors and RPCs distortion dependency.

Table 6: Principal point adjustment in pixel

PRINCIPAL POINT COORDINATES BEFORE AND AFTER INTRINSIC ADJUSTMENT

Components	Frame centre	Sat 1048	Sat 0f22	Sat 0f15	Sat 103c
x before	3300.	3309.4 (+ 9.40)	3288.23 (-11.77)	3319.51 (+19.51)	3348.14 (+48.14)
y before	2200.	2251.24 (+51.24)	2278.89 (+78.89)	2181.65 (-18.35)	2193.09 (- 6.91)
x after	-	3309.11 (+ 9.11)	3287.94 (-12.06)	3319.22 (+19.22)	3281.59 (-18.41)
y after	-	2278.2 (+78.20)	2306.19 (+106.19)	2207.78 (+7.78)	2187.79 (-12.21)

Hence, ASfM extracts correspondences from RPCs, creates physical models from RPCs too and refines them. These outcoming location models are accurate and they guarantee the epipolar constraint between images. The process continues with the dense matching part.

2.3. MULTI-SCENE STEREO (MSS)

2.3.1. PURPOSE

The last step of a Structure for Motion (SfM) process is the dense matching that Schonberger (2016) named reconstruction. Conversely to the bundle adjustment, all camera parameters are assumed perfect and fixed as starting criterion to the point triangulation by spatial intersection. Hence each point of an image matches its homologous pair in other images. The measured disparity (parallax) coupled with camera parameters are triangulated returning point object coordinates. It is the longest part because every scene pixel must be correlated to a large number of potential homologs in order to select the matching one, and this for all stereo pairs. Therefore, the reconstruction implementation matters.

The other important aspect of that step is the information redundancy. Former geometry assessments showed us that PS provides a vertical accuracy from 20 to 100 m (cross-track and in-track overlaps) at 4 m GSD. Multiple observations from several scenes set up a redundancy which improves the final accuracy after combination. These Multi-View Stereo (MVS) methods are a worthwhile point combination to improve the vertical accuracy, and thus the selection of the best method is discussed hereafter.

Finally, the process outcome shall be the expected DSM raster product with standard encoding. A consistent accuracy shall also be part of the file and inform of the product quality.

2.3.2. ALGORITHM DESIGN

As we saw in section 2.2, all camera parameters are consistent with each other after the bundle adjustment. We also chose to use physical models instead of mathematical ones allowing epipolar transformation. This is a homography which transforms the original scene into the epipolar plane of the stereo pair. That transformation can be derived from extrinsic parameters which fulfil epipolar constraints. On the other hand, mathematical models would require the similar transformation computed from new tie points, as presented in annex 3, changing global results from the performed block adjustment. Hence, the pairwise dense matching based on adjusted camera parameters returns consistent triangulated models from couple to couple. By extension, any Multi-View Stereo method requires a bundle block adjustment before.

Firstly, we focus on stereo pair matching. With ASP, the *stereo* function performs the dense matching. It uses adjusted camera parameters and scenes to compute epipolar images. They are distortion-free and normalised images in the epipolar plane in order to reduce the computation efforts of following steps. Then, seed points (tie points) are extracted along the epipolar axis (image x-axis) to bound disparities. The correlation process over subsampled images returns integer matches at sparse positions followed by the full resolution match within sparse boundaries. That full integer disparity is refined by subpixel interpolation fitting a parabola. Before the triangulation, the function filters these disparities according to their neighbour mean. The process ends with the creation of a point cloud encoded into a 4 bands tif file which is an ASP standard format.

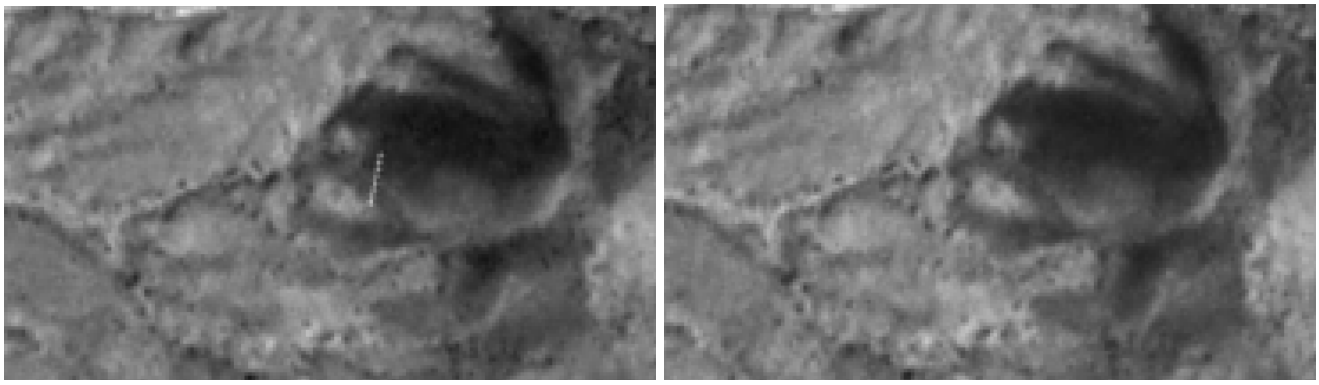
Nowadays, the Semi Global Matching (SGM) algorithm released by Hirschmüller *et al.* (2008) provides an efficient method for disparity extraction and remains a standard. The *stereo* function can run different correlation algorithms like SGM. We choose that faithfull method but a comparison with the More Global Matching (MGM) algorithm which is an improved SGM version released by Facciolo *et al.* (2015) would be interesting to assess because it may return less noisy disparities.

In order to accelerate the process and improve it, we take action before and after the ASP *stereo* function. The preprocessing from the *stereo* function (first part) is replaced by a handmade function called *EpipPreProc* based on OpenCV. The *stereo* function creates images through filters which make the process stable but longer. Due to the number of scene combinations in our blocks, a short spared time makes a large difference at the end. OpenCV provides a faster method transforming image matrices into the RAM. It requires a larger RAM space to load the full image but it divides the processing time by 12 on average. In addition, that function gives us access to image normalisation before dense matching (from unsigned 16 bits integer to [0,1] float values). The *stereo* function requires float images achieved with a 1st order polynomial application per pixel. Polynomial coefficients are computed per image histogram in order to balance light differences. Large differences appear with low orbit satellites with respect to higher orbit satellites, as we saw in section 1.2. In such a case, the information content of dark images (low orbit) is compressed to few bits leading to quantization defects and the histogram stretch would not rebuild the radiance range. An

attempt of image repair has been made using a Difference of Gaussian (DoG) filter which smoothes that noise and enhances the land cover sharpness.

Although, it still does not return good results due to wrong bundle adjustment with tie point lack and failing epipolar constraint. Rather, we prefer to include an additional filter about sun elevation at the scene selection stage, section 2.1. The use of the DoG filter over clear scenes would smooth low texture areas and lead to correlation issues. Furthermore, as displayed in the following figure, PS scenes may have hot pixels which come from satellite onboard reading mistakes. For instance, the left image contains a short bright strip in its middle that does not appear in the right image, sensed some days before. Any window convolution over hot pixels would spread out the error and block the correlation over larger areas.

Figure 12: Hot pixels of Dove-Classic, left: 20190831_180521_1040 (with), right: 20190813_180204_100c (without)

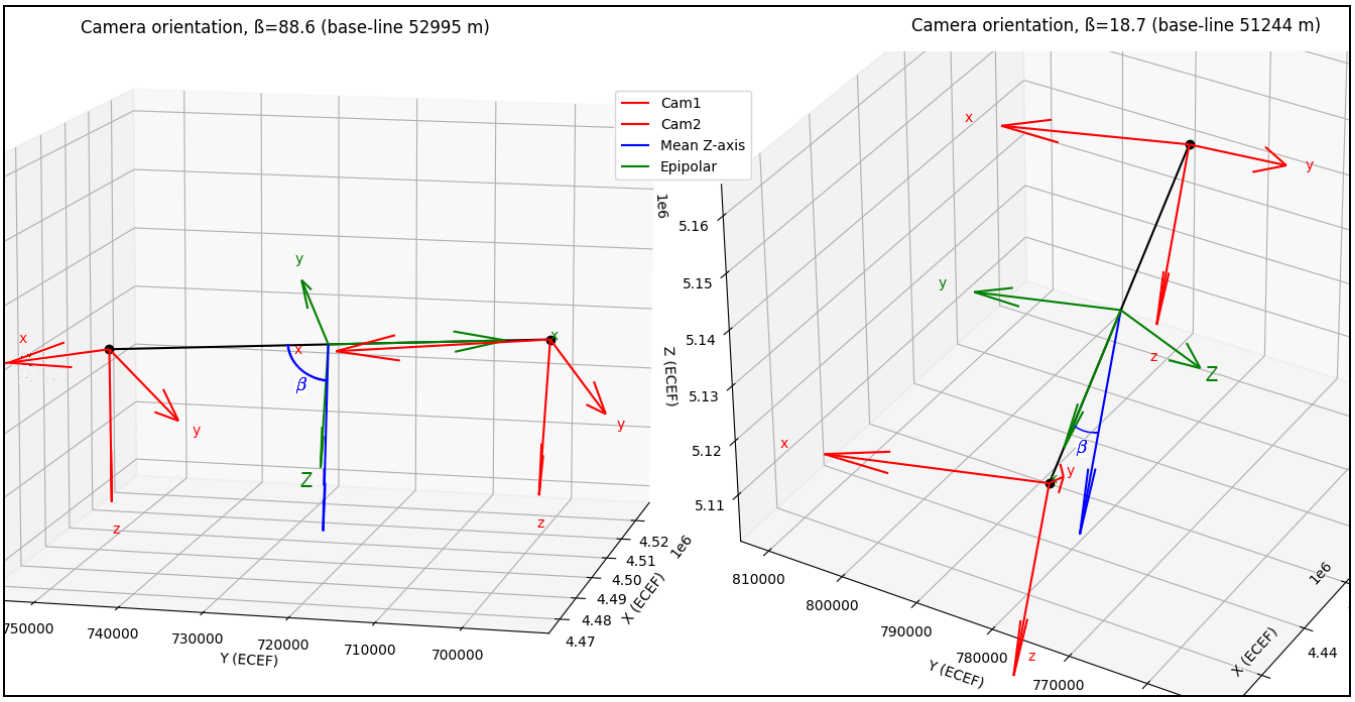


In addition, the *EpipPreProc* function applies an epipolar transformation. The *stereo* function is based on Fusiello *et al.* (2000) geometry and thus the *EpipPreProc* function computes the same epipolar geometry after distortion correction. The baseline extends along the scene x-axis and the average sight direction along the z-axis. Nevertheless, PS scenes may set non-suitable cases for a Fusiello transformation. As explained in the Fusiello's publication, the method fails whether the base-line is close to the sight axis and thus the epipole shows up within the homologous image. A similar case happens when PS satellites fly at different orbit heights, with the difference that sight axes are slightly different, sensing side to side scene footprints at 450 km height. In that case displayed in the following figure, the angle β between the baseline and the average sight direction is smaller than 80° and the Fusiello transformation points far away leading to largely distorted epipolar scenes. The *EpipPreProc* function cannot create those epipolar images due to the image size (from 28 MB to 19 GB), all the more that the correlation runs more time over a largely distorted image.

There exists several alternatives to ciruvent it. ASP is able to match images without epipolar transformation. The correlation process is longer and weaker against outliers because correlation windows slide in all directions in order to extract the correct match. Another way comes from mobile mapping experience and transforms the image to polar representation from the epipole. However, it requires changes in triangulation formulas that we leave to the *stereo* function. The most relevant choice would be to project the homologous image into the main image frame. It avoids the epipolar plane creation and it remains a fast run. Nevertheless, it has been late understanding that we could

not implement due to the time limit of the study. Therefore, we keep in mind that all large B/H stereo pairs cannot be computed yet.

Figure 13: Suitable Fusiello case with PInaetscope, left: suitable, right: non-suitable, red: camera orientation, blue: average sight direction, green: epipolar orientation.



During dense matching, ASP sets one image as reference (“left” image, first in command line) and finally, it filters out the disparity by Mean Absolute Distance (MAD). We improve the disparity filter by disparity comparison of the opposite match. We match the stereo pair with *“left”* image as reference and a second time with *“right”* image as reference. Then, we sum up reversed disparities to discard pixels with a disparity difference above 1 pixel while all other pixels hold the disparity mean. Thereafter, we let the triangulation part of the *stereo* function run. That method provides an additional filtering step which was not applied by the *stereo* function and it guarantees matched points.

ASP writes the dense matching result in a raster file with ground coordinates. That specific format describes points ECEF coordinates (band 1, 2 and 3) in the epipolar farme, named “*_PC.tif”*. It also records intersection errors from the distance in metres between intersecting rays in the 4th band. There is a *point2las* function converting “*_PC.tif”* file into point cloud *las* file but it skips the intersection error, so we replace it by a handmade function based on PDAL. As presented in the ASPRS documentation (2008), the *las* file is a LiDAR standard with fixed scalar fields shown hereafter. Therefore, the intersection error is recorded in millimetres in the “Intensity” field. We also record the incidence angle computed from the B/H ratio of the stereo pair in the “ScanAngleRank” field. The incidence angle in degree is rounded to fit signed 8 bits integers [-90, 90] but it still states the

geometry rank and can be used later on. We also write the stereo pair ID in the “PointSourceId” field encoded on 2 bytes allowing 65536 stereo combinations.

Table 7: *Las* format scalar field

Item	Format	Size	Required
X	long	4 bytes	*
Y	long	4 bytes	*
Z	long	4 bytes	*
Intensity	unsigned short	2 bytes	
Return Number	3 bits (bits 0, 1, 2)	3 bits	*
Number of Returns (given pulse)	3 bits (bits 3, 4, 5)	3 bits	*
Scan Direction Flag	1 bit (bit 6)	1 bit	*
Edge of Flight Line	1 bit (bit 7)	1 bit	*
Classification	unsigned char	1 byte	*
Scan Angle Rank (-90 to +90) – Left side	char	1 byte	*
User Data	unsigned char	1 byte	
Point Source ID	unsigned short	2 bytes	*

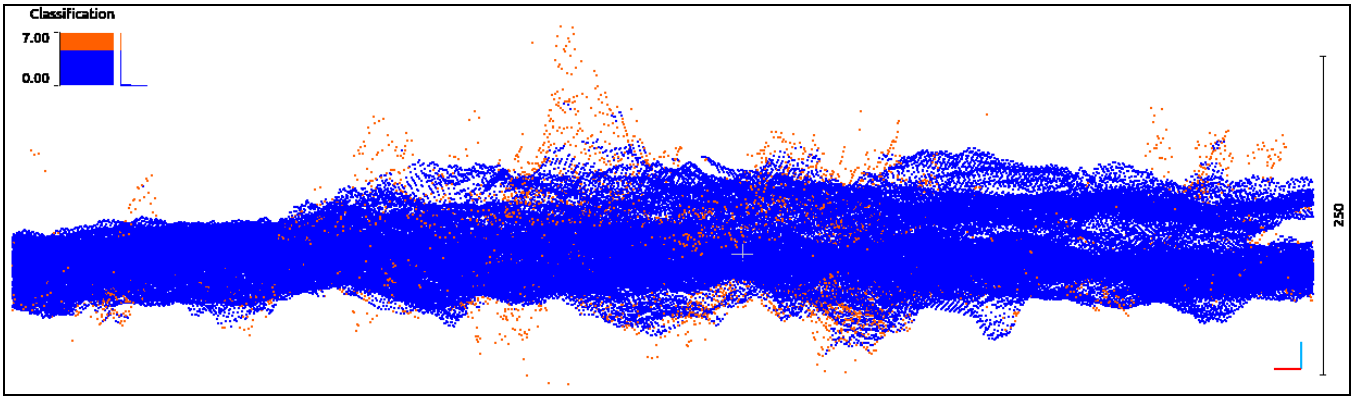
These pairwise dense matching may fail. The first reason is the overlap size according to the land cover. ASP needs seed points at an early stage and small areas or flat images (in terms of radiometry) do not provide enough points stopping the process. Regardless of the seed point number, a wrong bundle adjustment, due to missing tie points between scenes for instance, does not fulfil the epipolar constraint. Thus, the returned disparities are filtered out and the function returns an empty point cloud. The same consequence happens with noisy images where disparity peaks are filtered out by MAD. As we saw in section 1.2, images are noisier than usual with low orbit satellites making the sensing time in early morning. They deliver dark images with quantised radiometry failing the correlation.

As we saw in section 2.1, the SSBP part computes scene combinations based on vector footprint. That vector file presents stereo pairs as well as stereo triplets and more scene combinations. The scene couple reconstruction provides already a height redundancy which can be increased by triplet and more reconstruction using MVS method. The software ASP supports a MVS method at the triangulation stage, even though it is still an experimental implementation. It sets the first image as reference and computes its disparity against other scenes. Then, from the Slabaugh *et al.* (2001) publication and algorithm, the triangulation solves a least square between sight rays. We can interpret it as mean computation. Nevertheless, it appears that this result would be equivalent to the next part while it requires a lot of computation.

As a matter of fact, we use a similar MVS method on the difference that we run along raster cells, maximising redundancy, instead of stereo combination. The stereo pair clouds are merged into compilation tiles, named “Full”, with the PDAL library. That tiling process avoids computation burn with large files and a 10 m buffer avoids edge effect during following steps. At this stage, tiles include a height redundancy and a point quality value. Before the final rasterization which is our MVS core, we include cloud morphology information. As we can see in the following figure, there is noise due

to stereo geometry and scene quality that is not matching the overall cloud morphology. It is detectable with LiDAR classifying tools. Unlike LiDAR products, dense matching shall present a single ground surface and above or below points can be filtered out . Therefore, we apply a filter chain for every tile with Extended Local Minimum (ELM) filter presented by Chen *et al.* (2021) and a statistical outlier filter. These tools set point classes with respect to spatial position and distribution, using the class 7 (noise) as presented in the ASPRS documentation (2008). The ELM filter sorts points within cells in order to extract aberrantly low ones then classified as noise. It is defined by a cell size and a threshold value. That threshold is the length tolerance to the point above. A small value would be a strong filtering and we use a 1 m threshold value over a 10 m cell size discarding erroneous points. The statistical outlier filter computes a point distance threshold during a first run and selects outliers during the second run. The point distances to its k neighbours (8 in our case) provide the distance mean and the distance standard deviation. The filtering threshold is the sum of that mean and the standard deviation which can be scaled by a user factor that we set to 2. The outlier selection runs with respect to the computed threshold. Hence, they create a point quality redundancy.

Figure 14: *Full* point cloud with matching noise, blue: unclassified, orange: noise class from morphology filter



Finally, point cloud tiles present redundancy about height and about point quality. This last step turns those redundancies into a lighter product with improved accuracy. A rasterization yields a DSM grid lighter than point clouds and allows point average. It is commonly known that elevation products should be sampled over 3 times larger cells than the original GSD. Hence, a 4 m scene (GSD) would lead to a 12 m DSM (GSD). Although, the large information redundancy available with PS scenes enables GSD reduction to the original one. As we see in the next section 3, 1 time the original GSD already means a large number of points per cell. A higher GSD factor would improve the outcome but it would smooth the height variation. We choose to not subsample our DSM product (GSD factor equal 1) in order to draw up the achievable accuracy without smoothness factors. The final product records the height and its corresponding accuracy. It also records the number of averaged points as additional information. In section 3, we compare height with respect to the ground truth and link it to the computed accuracy.

Instilled by different MVS methods like Slabaugh (2001) publication or ARGANS works with Satellite Derived Bathymetry (SDB), we figure out a reasonable merge. A point mean along the DSM grid

would be the simplest MVS version, providing one elevation value and a Standard Error of Mean (SEM) without taking into account the recorded accuracy. A median would be stronger against outliers but does not still take the advantage of the known accuracy and LiDAR filters. Therefore, the best option remains a weighted average with the inverse point accuracy as weight. As presented in annex 6, we consider the intersection error as point variance in the horizontal plane. After transformation into the vertical direction, the point mean can be weighted by inverse variance. We discard noise points from morphology classes. During the gridding, the Inverse Distance Weighting (IDW), also known as Shepard's method, is another available option, in place of arithmetic mean. We prefer the box filter instead because IDW method overweight close points and returns spikes. However, it would be equivalent to a subpixel refinement and an assessment in the case of very high redundancy, not discussed here, should present promising results.

Table 8: Final product description

Digital Surface Model (DSM)	Band 1: float 32 bits, "Height" (merged height $\overline{Z}$) Band 2: float 32 bits, "Accuracy" (height accuracy $\overline{\sigma_H}$) Band 3: float 32 bits, "PtCount" (point count N)

3. RESULTS

Afterward, that methodology runs over 3 Area Of Interest (AOI) as test procedures. Outcoming results are compared to the ground truth. Image blocks are selected in a close time period to the ground truth survey. We run several month blocks per test site with a redundancy of 2, meaning that we select the 3 best stereo pairs over each AOI point. The next AOI sections start with a process summary table with some key values from the computation. They end with an assessment table with accuracy information. In between, we discuss specific outcomes.

Accuracy tables present statistical metrics from the difference between the ground truth and the computed DSM (dGT) like the mean, the standard deviation, etc. That average means the product bias, the 90th percentile comes from the boundary of the cumulative absolute histogram and the Normalised Median Absolute Deviation (NMAD) provides a robust estimator from the Höle *et al.* (2009) work. In addition to them, it includes metrics from the computed accuracy (band 2), as presented in annex 6. The standard deviation, the NMAD, the 90th percentile and the maximum are expected equal to dGT ones in case of consistent formulas. However, its average outlines another characteristic than dGT mean and either presents the error average of a product.

3.1. Providence Mountain, USA (California)

Providence Mountain is the development test site and we restart its process afterware. This block holds a large height range over a steady land cover. However, It is a much larger site than the others and its process remains long. We start with 3 scene blocks from the same period as the survey acquisition: August, September and October 2019.

The LiDAR ground truth is reprojected into UTM 11 N projection and the ellipsoidal height is retrieved by sum with the EGM08 geoid as explained in the NAVD88 height system description.

Table 9.1: Providence block process summaries

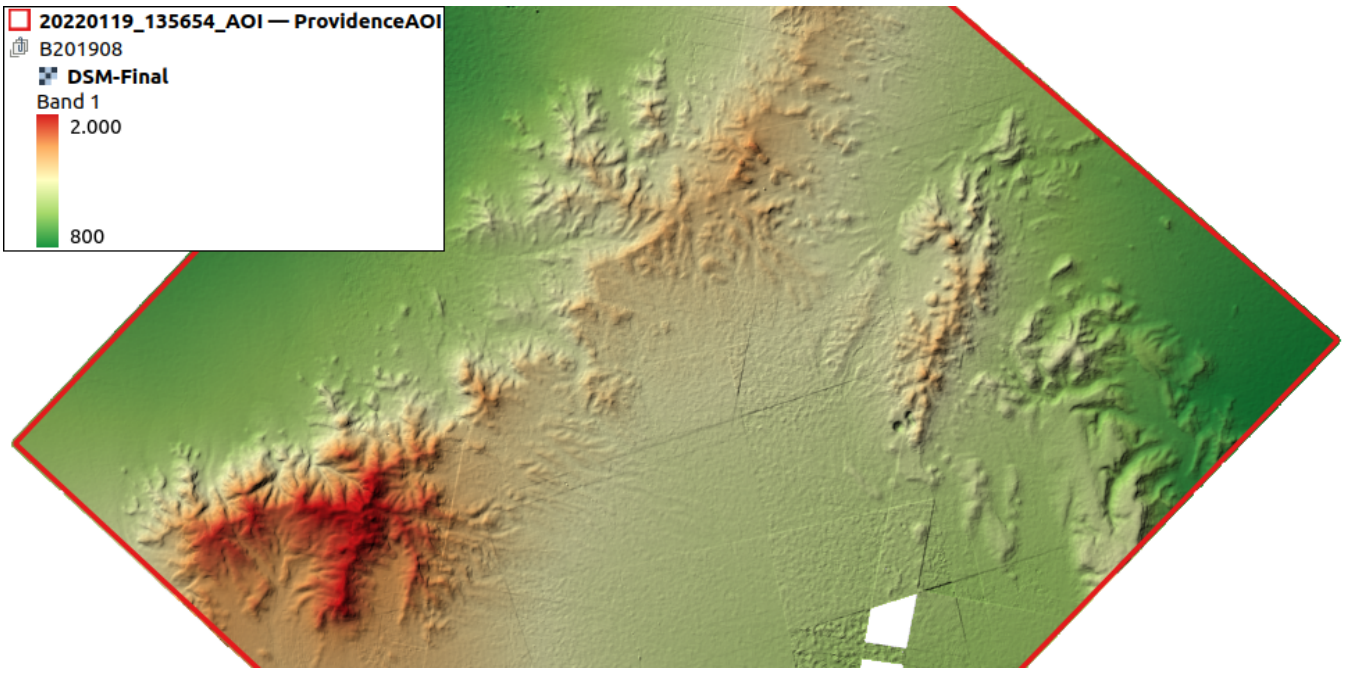
Block ID	Scene number	Maximum overlap	BA residual means position, orientation, key points	DSM, minimum redundancy
B201908	128 (8.4 GB)	18 scenes	0.805 m, 0.123 °, 0.48 pxl	0 stereo pairs (hole)
B201909	109 (7.1 GB)	15 scenes	0.279 m, 0.075 °, 0.12 pxl	⊗
B201910	127 (7.9 GB)	16 scenes	0.668 m, 0.118 °, 0.44 pxl	0 stereo pairs (hole)

Due to the AOI size, the number of selected scenes is larger than the others and thus makes processes longer. Besides, these blocks use scenes created before summer 2020. There was a process change at that period and older scenes come with different metadata than our development experience. Thus we included some changes during the block creation (SSBP).

For process reason, an error appended during the block B201909 computation. Key points were extracted using the image extraction, as explained in section 2.2. Its consequence was a much higher number of points (around 150 000 while 50 000 in other blocks) spreaded into clusters with just a few multi-observation points. Hence, the bundle adjustment overfits those points without consistency and stemming wrong epipolar constraints. So, we stop that block process.

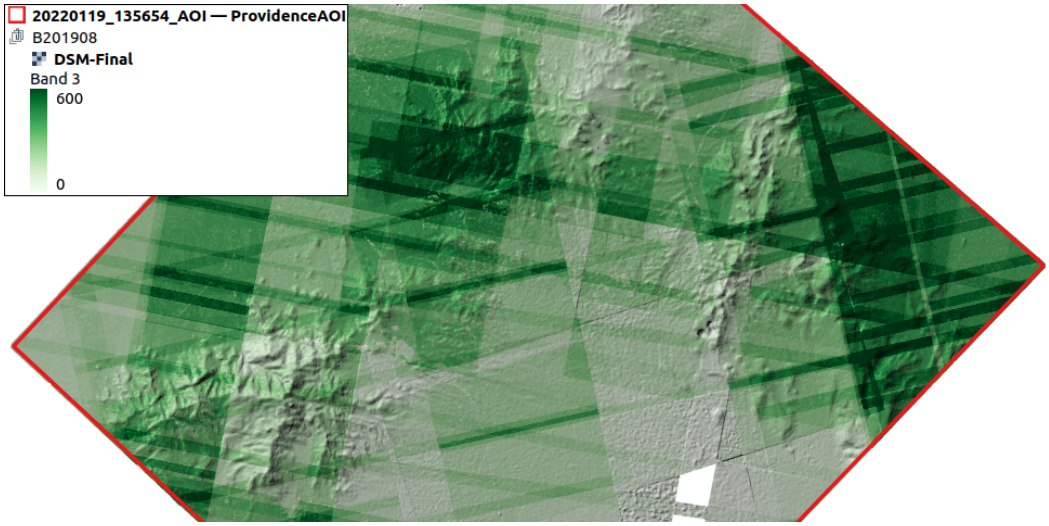
The process of the block B201908 went to its end and returned good results as we can see in the following figure. We recognize mountains, hills, flat areas and plateaus. The height variation is in the correct order of magnitude. Although, we observe holes in the product at the bottom due to the lack of stereo pair. That part is only filled with large B/H ratio couples selected during the block creation and no default redundancy exists there by chance. Hence, the dense matching part (MSS) skipped tough matches due to the epipolar image size and the effective matching (MSS) does not correspond to block descriptors (SSBP). Furthermore, we observe rough tiles and steps in the central part also related to the lack of redundancy and remaining errors in the bundle adjustment.

Figure 15: Providence B201908 height result, DSM extract of “Height” band (1)



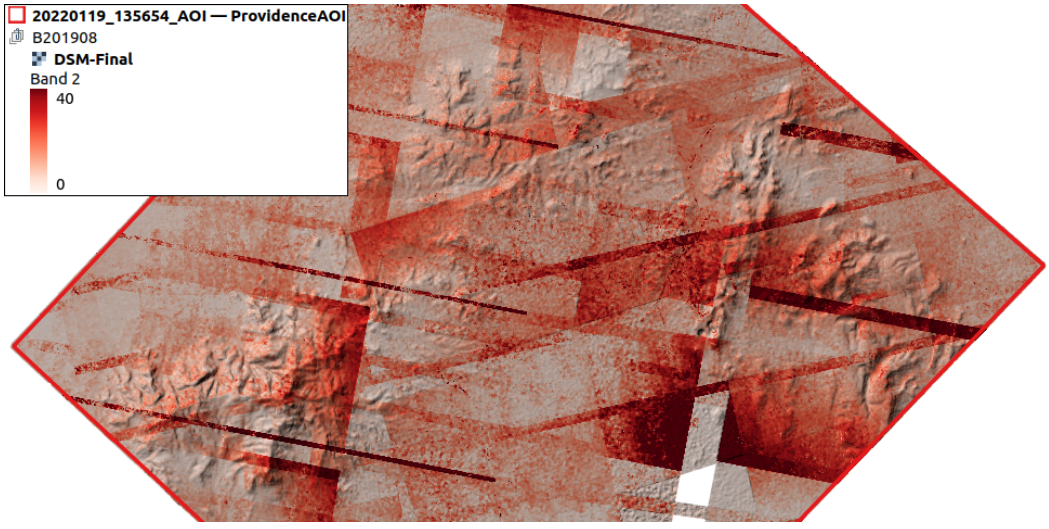
The 3rd band of the outcoming product displays the number of points in weighted averages. As discussed in section 2.3, the number of points is already sizable at the original scene GSD reaching 600 merged points per cell. Despite the coverage hole and the redundancy lack, there are 263 points in average per cell (standard deviation 156) making our MVS stable against outliers. Later on, any improvement of the matching part (MSS) shall return more point clouds and improve results.

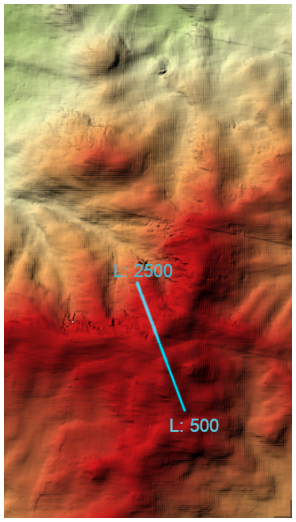
Figure 16: Providence B201908 count result, DSM extract of "PtCount" band (3)



In the 2nd band, the merging method records the elevation accuracy, as explained in annex 6 and presented in the following figure. We find again rough tiles quoted before with higher accuracy values which go until 40 m, demonstrating the formula consistencies. However, there are accurate rough areas, next to holes, because they come with only one stereo pair in the computation. The vertical standard deviation ($\overline{\sigma_Z}$) is added to a null point standard deviation (σ_P). This weakness is still related to the lack of redundancy yet.

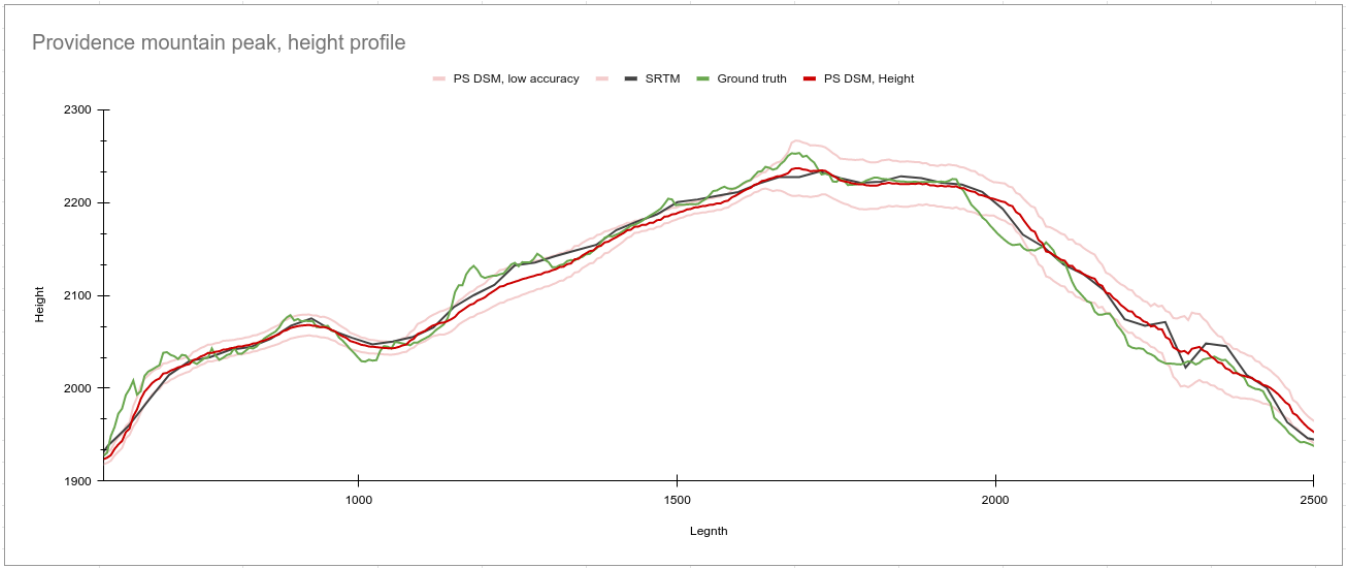
Figure 17: Providence B201908 accuracy result, DSM extract of "Accuracy" band (2)





In addition, we compare the product height variation to the ground truth at the mountain peak. The height profile in the next figure comes from the blue line plotted beside. The profile shows the SRTM elevation (black), the ground truth (green) and the product height (red). This last one is buffered by the computed accuracy providing the validity range around (light red). So the vertical comparison focus of model similarities and inclusion of ground truth within the validity range. We observe a close similarity between the product and the ground truth as well as an incorporation of that ground truth in the accuracy buffer in most of the cases. The peak height differs by 16 m but the ground truth peak remains in accuracy range. That peak difference is related to local features (rocks) that are partly visible at scene GSD and could not be modelled in a 30 m GSD SRTM. Similarly, missing height frequency variations are due to sampling differences. As a whole, the mountain modelling is ensured by the process.

Figure 18: B201908 height profile at the Providence Mountain peak, black: SRTM (30 m GSD), green: ground truth (1m), red: B201908 DSM (4m)



The process of the block B201910 returns even better results than the previous one. It is not complete but its redundancy coverage shows more stereo point clouds. Its accuracy assessment, available in the accuracy table hereafter, displays these improvements.

Table 9.2: Providence block accuracies, dGT: ground truth comparison, Band 2: computed accuracy

Block ID	Mean (dGT)	Mean (Band 2)	Standard deviation (dGT / Band 2)	NMAD (dGT / Band 2)	90th percentile (dGT / Band 2)	Absolute maximum (dGT / Band 2)
B201908	-6.2 m	13.7 m	11.9m / 12.3 m	9.7 m / 6.8 m	20,6 m / 25,0 m	206.8 m / 2047.5 m
B201910	3.2 m	7.9 m	8.3 m / 10.7 m	7.6 m / 2.9 m	12.2 m / 15.5 m	229.6 m / 129.8 m

3.2. Stuttgart, Germany

Stuttgart is a temperate region in a small height range. The AOI includes the city, agricultural fields and forest which have different image patterns than the previous site. The image pattern affects the key point extraction and the dense matching. Conversely to desert landscape, agricultural field radiometry changes in a short period of time as well as temperate forests. We run the algorithm along 5 month blocks from the survey period: from May to September 2020.

The LiDAR ground truth is reprojected from the GRS 80 ellipsoid based system (ETRF 89) to the WGS 84 one (ITRF), both projected by UTM 32 N. The German survey institute provides altitude DSM using the German height system DHHN2016 based on the quasi-geoid GCG 16. The difference with the EGM08 appears to be around 12 centimetre as shown by the BKG online tool. Therefore, we use the EGM08 to extract the local ellipsoidal height since we expect a larger accuracy.

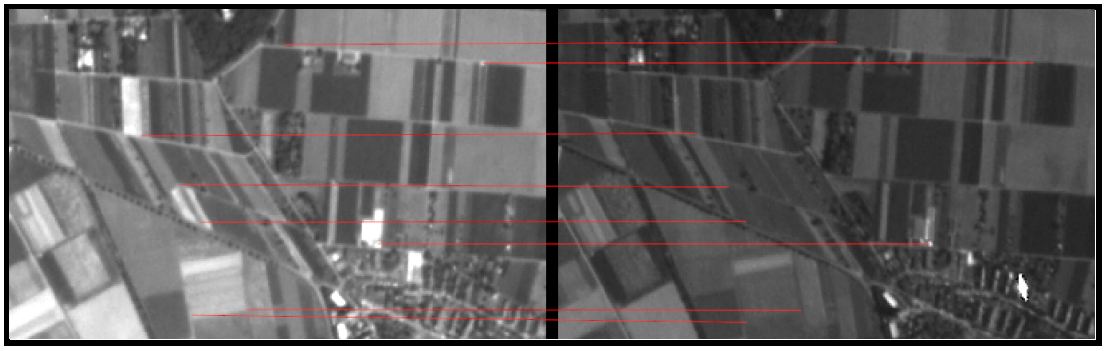
Table 10.1: Stuttgart block process summaries

Block ID	Scene number	Maximum overlap	BA residuals mean position, orientation, key points	DSM, minimum redundancy
B202005	17	⊗		
B202006	16	⊗		
B202007	29 (2.0 GB)	9 scenes	0.528 m, 0.102 °, 0.43 pxl	1 stereo pairs
B202008	39 (2.6 GB)	10 scenes	0.525 m, 0.090 °, 0.44 pxl	2 stereo pairs
B202009	39 (2.5 GB)	11 scenes	0.413 m, 0.128 °, 0.45 pxl	3 stereo pairs

After strict filtering during the block creation (SSBP), blocks B202005 and B202006 cannot be filled up with enough scenes. This due to the weather of that period and the cloud limit filter involved in API query. Thus, we discard these 2 blocks.

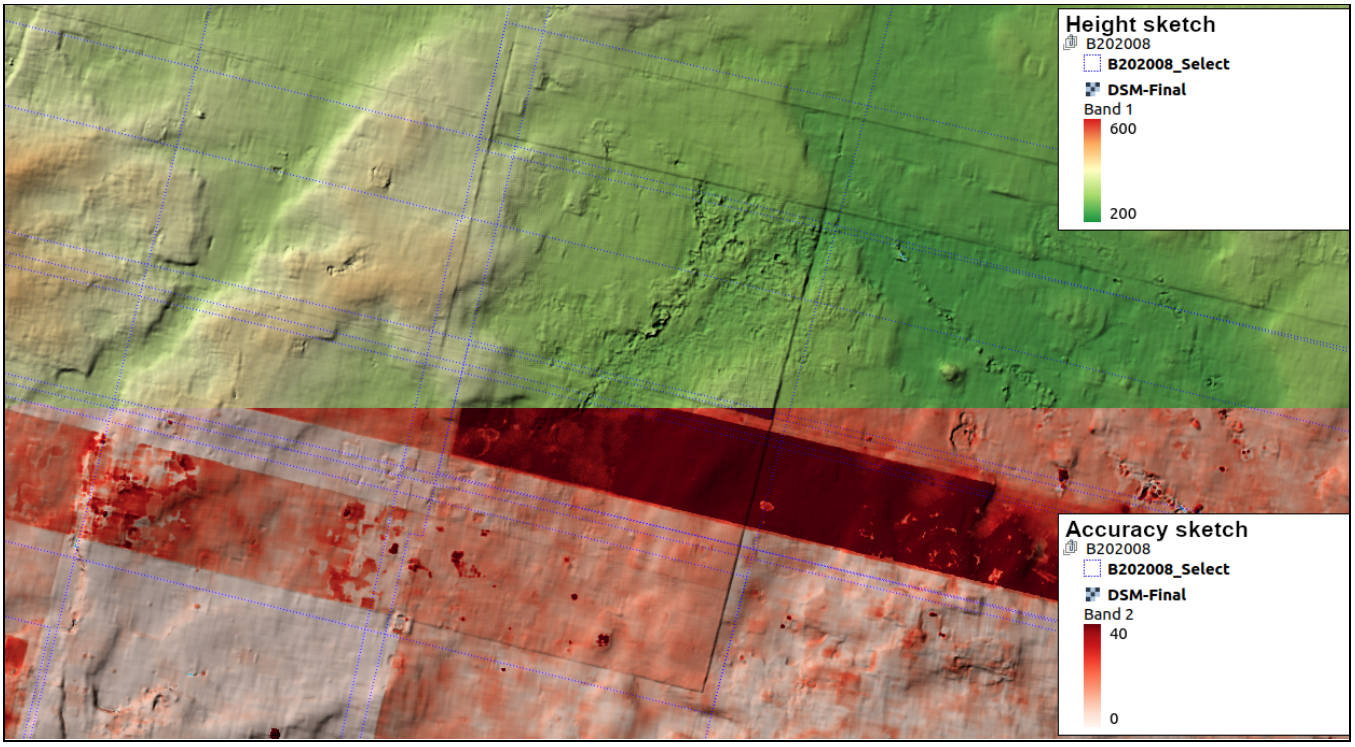
During the process of these blocks, we improved the tie point filter during the extrinsic bundle adjustment. Matched key points from grid extraction tied multiple scenes and the first EO-BA iteration adjusts extrinsic parameters as well as tie point coordinates (image and object space). Before the second iteration, some are filtered out with respect to their reprojection error. It was originally fixed at 80% of them below 2 pixels and remaining ones up to 3 pixels, and we changed to 90% up to 1 pixel and the remaining ones until 2 pixels. We do so because agricultura fields display sharp corner points but they are not stable over a few days. The following figure displays manual matches (red) in images from close dates (5 days). Hence, wrong matches over changing land cover pop up during the first iteration and are discarded at the second iteration.

Figure 19: Land cover changes in Stuttgart, left: 30th Jul. 2020 scene, right: 25th Jul. 2020 scene, red: manual link of failing correspondances



The results of B202008 displays the importance of that bundle adjustment. As shown in the next figure, we observe steps in the final DSM (top part) corresponding to scene footprints (blue dots). Failing scene coregistration makes stereo couples inconsistent. The merge process (MSS) averages all points according to the computation geometry and the intersection error without taking into account any bundle adjustment error. Nevertheless, as we can see in the bottom part of the figure (with continuity), the height accuracy band displays that misalignment from the point standard deviation at the rasterization stage. Hence, the bundle adjustment was assumed to be correct but the final product displays remaining errors which affect final statistics, as displayed in the accuracy table.

Figure 20: DSM steps in block B202008, continuity between top and bottom parts, top: DSM elevation, bottom: height accuracy



The block B202009 is the best process of that AOI. It returns a consistent model with a small accuracy. Its comparison to the ground truth presents a similar assessment. A close look at the building area presents the PS ability to recontonstruct small feature elevation. In the following figure, we compare the DSM over the EnBW tower and the Mercedes stadium with the ground truth and the SRTM. Hence, we can see building elevations while it was missing in the SRTM. However, PS GSD is not able to recreate the full tower height nor both stadium sides due to sampling limits.

Figure 21: Height profile from B202009, results through EnBW tower and Mercedes Stadium

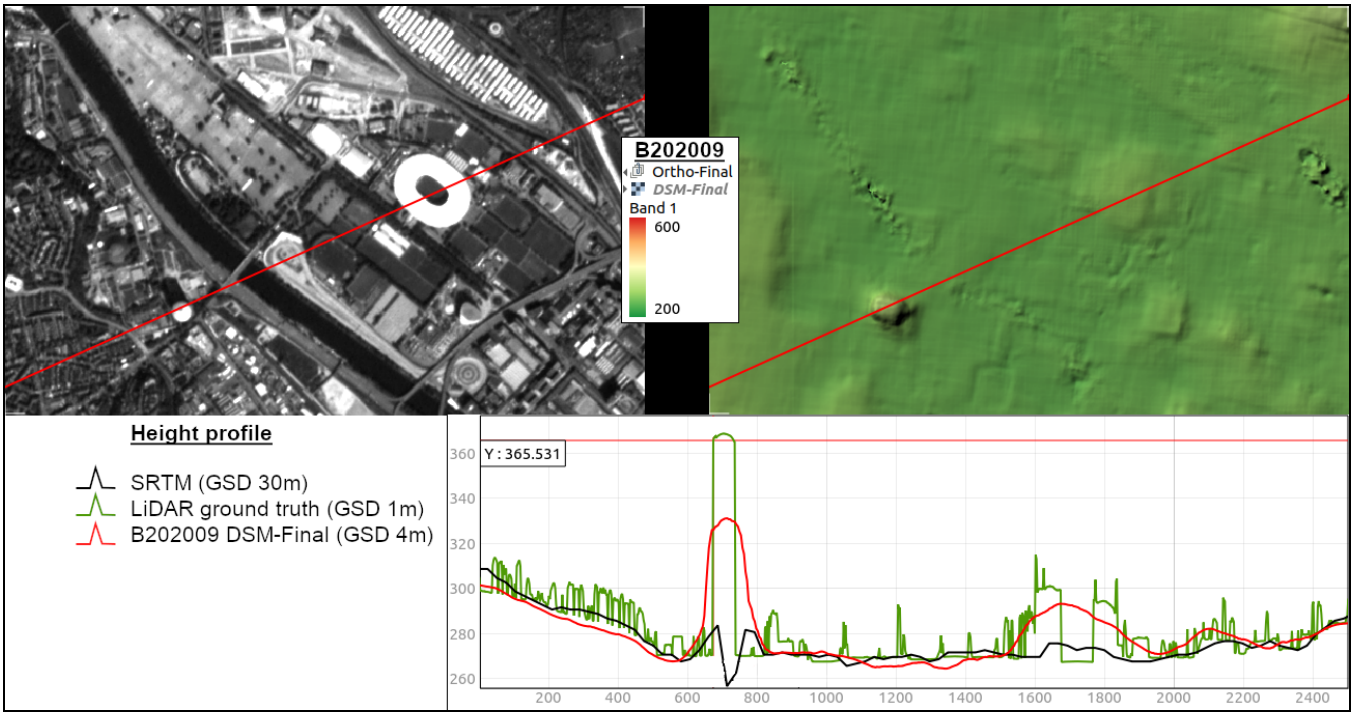


Table 10.2: Stuttgart block accuracies, dGT: ground truth comparison, Band 2: computed accuracy

Block ID	Mean (dGT)	Mean (Band 2)	Standard deviation (dGT / Band 2)	NMAD (dGT / Band 2)	90th percentile (dGT / Band 2)	Absolute maximum (dGT / Band 2)
B202007	23.8 m	24.5 m	42.0 m / 67.5 m	24.8 m / 11.6m	53,9 m / 38,9 m	2328.7 m / 1369.9 m
B202008	10.7 m	14.1 m	12.0 m / 16.8 m	13.0 m / 7.6 m	25,4 m / 27,9 m	708.6 m / 748.7 m
B202009	8.9 m	12.1 m	9.5 m / 15.4 m	9.6 m / 5.5 m	21,8 m / 22,3 m	988.0 m / 596.3 m

3.3. Mount St Helens, USA (Washington)

Mount Saint Helens is also a temperate region with different landscapes than before. The high volcano, in the middle of the AOI, is under a snowcap in winter leading to bright reflectance and even scene saturation. A dense pine tree forest surrounds the peak which does not change along the year but shows a complicated image pattern. Moreover, there is a lake at its North-East corner where we do not expect consistent results. We start with 3 scene blocks from summer 2019: August, June, July and August 2019.

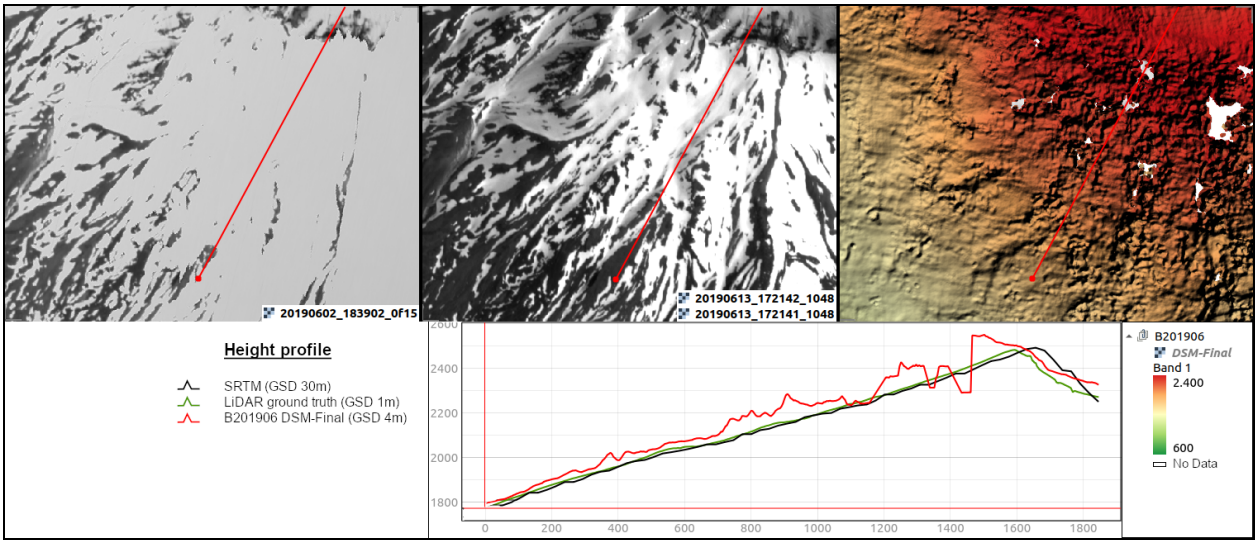
Like Providence Mountain, the LiDAR ground truth is reprojected into UTM 10 N projection and the ellipsoidal height extracted with the EGM08 geoid.

Table 11.1: Mt St Helens block process summaries

Block ID	Scene number	Maximum overlap	BA residuals mean position, orientation, key points	DSM, minimum redundancy
B201906	32 (2.1 GB)	14 scenes	0.791 m, 0.094 °, 0.50 pxl	3 stereo pairs
B201907	32 (2.0 GB)	14 scenes	0.565 m, 0.100 °, 4.40 pxl	⊗
B201908	38 (2.3 GB)	13 scenes	0.825 m, 0.108 °, 0.49 pxl	3 stereo pairs

Regardless of the ground truth survey from August 2018 to April 2019, we choose image blocks during the following summer avoiding snowcaps. Despite that choice, the early scene of the first block (B201906) still contains snow which melts during the month. Hence, the block includes saturated scenes and changing land cover. It leads to the same key point errors as seen at the previous AOI. The shrank outlier filter helps again the bundle adjustment. However, changing land cover troubles correlations and returns noisy point clouds. The following figure displays the B201906 result into a height profile with respect to 2 orthophotos. We figure out the melting speed between from 2nd to the 13th of June and observe chaotic results and even no-data area from failing matching.

Figure 22: Height profile from B201906, results through snowcap along volcano slope



During the scene selection (SSBP) of the second block (B201907), the B/H ratio filter fosters wide incidence angle and thus includes oblique scenes. As we see in the next figure, the 2 yellow footprints follow neither an ascending nor a descending orbit direction and their metadata records a 10° view angle (from nadir). It happens that the operation team at Planet tasked oblique acquisitions as explained in section 1.2. So, 2 scenes appeared by chance in that block selection. The usage of such images creates a better intersection at the ground providing accurate points if their registration succeeded. The following figure displays a 3D graph which presents camera centres at the initial stage (red triangle) and after extrinsic adjustment (orange dots). The view direction is drawn by arrows (red or orange). A scale factor applies on the length between initial and adjusted position changing actual coordinates. The implemented tie point method created matches in relation to these 2 scenes and our adjustment ran through. However, we observe very large key point residuals in the process summary table indicating adjustment error. We can conclude that our bundle adjustment method (ASfM) cannot handle large sight differences and we stop the process of that block here.

Figure 23: Oblique scene footprints (20190724_172037_0f24, 20190724_172038_0f24)

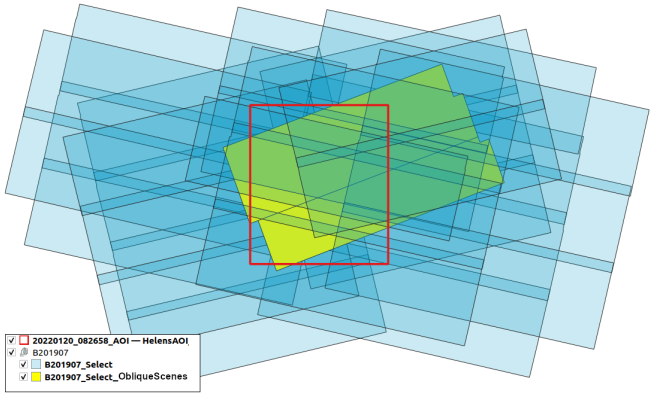
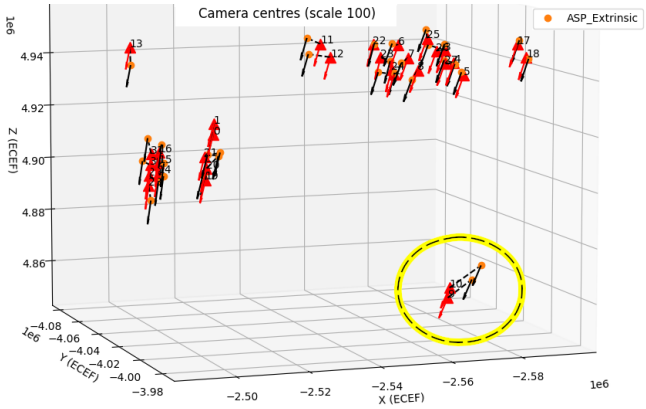


Figure 24: camera centres of B201907 block, yellow: oblique scenes



On the opposite, the block B201908 process used cleaner images and it provides better bundle adjustment results (ASfM) and a higher stereo pair redundancy. That yields the best result computed over Mount Saint Helens AOI.

Table 11.2: Helens block accuracies, dGT: ground truth comparison, Band 2: computed accuracy

Block ID	Mean (dGT)	Mean (Band 2)	Standard deviation (dGT / Band 2)	NMAD (dGT / Band 2)	90th percentile (dGT / Band 2)	Absolute maximum (dGT / Band 2)
B201906	-6.3 m	21.9 m	28.3 m / 42.3 m	15.7 m / 6.2 m	31,4 m / 41,1 m	1787.4 m / 738.3 m
B201908	5.3 m	21.4 m	42.8 m / 16.6 m	13.9 m / 8.1 m	26,6 m / 42,8 m	3825.6 m / 1816.9 m

CONCLUSION

As a conclusion, we retrieve pros and cons of method parts as well as pros and cons of a SfM process using Planetscope scenes. All in all, a photogrammetric process is possible from the PS setting and its redundancy enhances results. That first Planetscope SfM research using image based approach proves that Dove-Classic senses processable scenes acquired with enough disparity (parallax) in-between. Our block selection contains stereo pairs with large incidence angles which would benefit to the final result. The match accuracy is improved by the Dove's time resolution which allows redundancy. In our cases, we achieved a 8.3m accuracy (standard deviation) DSM with a 3.2m bias (Providence block), a 10m accuracy (standard deviation) DSM with a 10m bias (Stuttgart block) and even more could be achieved with some process changes.

The first part of our process (SSBP) returns a suitable way to extract scene blocks. The database parsing is efficient over several terabytes of raster files. This is counting on available metadata already computed during scene integration. However, the metadata discontinuity due to several versions of internal chains leads to many exceptions and makes the data parsing complicated in case of older (or newer) scene seeking. Moreover, the current version of our algorithm uses a limited number of strict filters during the API request but no more is taken into account for time limitations. It exists more and one shall benefit from the inclusion of Signal to Noise Ratio (SNR) criteria, for instance, in order to focus our chain on faithfull images. It is the case of the sun elevation filter included at a later stage in relation to low orbit satellites (section 1.2). From that query information, we slip scene lists into month blocks. The daily time accuracy of PS supplies enough scenes within 30 days. The scene selection proceeds with the powerful B/H ratio filter. From a short process time, it returns a sufficient number of scenes covering the whole AOI. However, satellite positions are recorded at the beginning of the Planet internal pipeline from aboard navigation instruments that are inaccurate. Hence, all pledged B/H ratios at that stage are much different from final camera positions. As we saw in section 2.3, there are more stereo cases than the first assessment in section 1.2. There are still in-track, cross track and cross date overlaps which depend on the satellite orbit height cases, the orbit direction and view angles. Our first version of the dense matching chain is not stable enough for large cases and thus we miss some expected stereo pairs. Hence, there is a difference between descriptor files presenting a large redundancy and actual point cloud results leading to lack of redundancy (only 1 stereo pair) and even holes in our final product. One shall improve the dense matching part (MSS) in order to reduce that difference. The main SSBP correction is either its conversion from a filter approach to a cluster approach. The feature space dimensions is the picky component but one would gain the entire redundancy available instead of discarding scenes.

The second brick about bundle adjustment (ASfM) returns consistent results, even though it requires processing time. We found in Providence Mountain tests that the selected key point extraction method is a must to tie multiple scenes. There are not enough multi-observation points through image based extraction while the grid extraction yields a sufficient number and a stiff block. Hence, it can rely on the input camera positions for in-air adjustment. Furthermore, ASP provides an engine using ECEF cartesian system which avoids any local projection issue. Nevertheless, we preferred to include some key points, from RPC extraction, fixed at SRTM height to avoid the block drift. These

ground control point with large accuracy $(\sigma_{X,Y,Z} = 10[m]; \sigma_{x,y} = 1[pxl])$ surround blocks and anchor it. Hence, the SRTM height is the only external data source. Regardless of the tie point extraction, the usage of RPC location models is replaced by Physical Models (PM) for dense matching reasons explained in section 2.3. The PnP method converting from one to another is consistent and any differences are corrected as well as misalignments by an extrinsic bundle adjustment. The intrinsic refinement in the condition of robust extrinsic parameters is legitimated by the Planet calibration method presented in section 1.7. Therefore it guarantees consistent scene blocks which fulfil epipolar constraints. Although, one key part of the ASfM part is the final criteria resuming the block relevance at the end. The final controle of a bundle adjustment remains important and it has been done manually during our tests. No final criteria has been designed and one may do it using the adjustment residuals and evolving statistics.

Finally, the reconstruction part (MSS) loops dense matching processes per stereo pair. The epipolar geometry and all tips around the original ASP *stereo* function in-use make the computation faster, even though it remains long (7 hours with 29 scenes, Stuttgart B202007). These also ensure output correctness by disparity comparison and intersection error recording. However, the current implementation cannot handle some incoming couples like satellites from different orbit heights or large differences in orbit direction. These specificities from PS flight bring large epipolar images which are not loadable into the memory. One shall implement more stable (but still fast) functions for pre-processing and disparity merge. Those stereo pair clouds are then filtered using LiDAR tools. They provide classifying methods based on cloud morphology and bring a point quality redundancy. There are currently 2 filters included but one could use more like Simple Morphological Filter (SMRF) in order to avoid matching defects. Inspired by several Multi-View Stereo methods, we came up with a weighted average method merging height redundancy. It also computes the height accuracy, also written into the DSM file. The DSM GSD fixed to the original scene resolution ensures the number of points per raster cell but one could implement a subpixel creation with other mean methods in case of sufficient redundancy.

Once the algorithm is completed, we run it over 3 test sites. They provide results from several blocks over 3 applications with different specificities like height ranges, land covers, changes, clouds, areas, etc. They state the achievable accuracy and draw out the remaining weakness. Hence, the 100m geometrical vertical accuracy from in-track overlap reduces to 10m after MSS and makes the product suitable for external release. All process defaults are assessed and presented in section 3.

Hence, the current method can handle Dove-Classic scenes for Digital Surface Model (DSM) production. The following development shall focus on Dove-R and Super Dove cameras. As we saw in section 1.3, Super Doves are wider cameras for a better photogrammetric setting at in-track and cross track overlaps. They use push-frame sensors and some changes have to be included in key point extraction and dense matching in order to manage subframes as independent images and avoid correlation from different colour channels. The push-frame sensor avoids Bayer interpolation, sensing all pixels per band leading to better results. Moreover, as we saw in section 1.4, there are more sensed images but only few are anchor frames coming with accurate location models. So the bundle adjustment must properly weight camera parameters according to the data source. We saw during the St Helens test, in section 3.3, that oblique images bring accurate geometry for dense

matching, even though our algorithm cannot handle them. During that thesis period, a set of oblique images was sensed over Providence Mountain from all Dove generations with large view angles, as explained in section 1.2. It leaves an opportunity for more work with matching engines able to tie oblique scenes. Another improvement, can be the creation of coloured point clouds that could replace the orthophoto Planet pipeline in the future. At least but not last, the current work speaks of DSM production from scene block handling intrinsic redundancy. One may design cross block merge or comparison. Hence, all processed blocks from one season could be merged into an enhanced result and time series comparison should show morphology changes. It shall bring a powerful tool for ground monitoring.

ACKNOWLEDGEMENT

Throughout the development and writing of this research, I have received great deal of support and assistance.

I would first like to thank my supervisor, Professor Norbert Haala, whose expertise was invaluable for research ability, methodology and formalising that thesis. I learned from his scientific rigour and broad approach bringing my work to a higher level.

I also would like to acknowledge all members of the Institute für Photogrammetry at Stuttgart led by Prof. Dr.-Ing. Uwe Sörgel for their help and support.

As an intern, I would like to acknowledge the Planet Labs company who offered me an inside position, without which it would not be possible to carry that study out. I thank Kelsey Jordahl as the director of Derived Data Products who supported me. I also thank the AQuA team from the German office led by Isabel Gonzales and Matthias Kolbe, more recently, for their wonderful collaboration as well as the Rectification team from the US office led by Brian Brown. I would particularly like to single out Duy Nguyen for his close support and Seth Price for his shared insight.

Finally, I would like to personally thank Antonio Martos as a former computer vision scientist at Planet who had faith in that research and shared his thoughts about it. His invaluable knowledge of Planetscope and all challenges brought by such computation guided me all along the study.

REFERENCES

S. Aati, J.-P. Avouac, 2020, Optimization of Optical Image Geometric Modelling, Application to Topography Extraction and Topographic Change Measurements Using PlanetScope and SkySat Imagery, Remote Sensing, December 2020.

ARGANS ltd, Satellite Derived Bathymetry and redundancy enhancement from Sentinel-2 images, https://sdb.argans.co.uk/.

ASPRS documentation, 2008, LAS Specification Version 1.2. approved by ASPRS Board 09/02/2008, https://www.asprs.org/a/society/committees/standards/asprs_las_format_v12.pdf.

S. Bhushan, D. Shean, O. Alexandrov, S. Henderson, 2021, Automated Digital Elevation Model (Dem) Generation From Very-High-Resolution Planet Skysat Triplet Stereo And Video Imagery, ISPRS Journal of Photogrammetry and Remote Sensing, January 2021.

Bundesamt für Kartographie und Geodäsie (BKG), Online calculation of quasi-geoid heights with the GCG 2016, http://gibs.bkg.bund.de/geoid/gscomp.php?p=g.

D. C. Brown, 1966, Decentering Distortion of Lenses.

Z. Chen, 2012, Upward-Fusion Urban DTM Generating Method Using Airborne Lidar Data, ISPRS Journal of Photogrammetry and Remote Sensing, August 2012.

G. Facciolo, C. De Franchis, E. Meinhardt, 2015, Mgm: a significantly more global matching for stereovision, In Proceedings of the British Machine Vision Conference (BMVC), BMVA Press: 90.1-90.12, September 2015.

C. S. Fraser, 1997, Digital Camera Self-Calibration, ISPRS Journal of Photogrammetry & Remote Sensing, March 1997.

C. S. Fraser, H. B. Hanley, 2003, Bias Compensation in Rational Functions for Ikonos Satellite Imagery, Photogrammetric Engineering & Remote Sensing, January 2003.

A. Fusiello, E. Trucco, A. Verri, 2000, A compact algorithm for rectification of stereo pairs, Machine Vision and Applications 12: 16–22, March 2000.

S. Ghuffar, 2018, DEM Generation from Multi Satellite PlanetScope Imagery, Remote Sensing, September 2018.

H. Hirschmüller, 2008, Stereo processing by semiglobal matching and mutual information, IEEE Transactions on Pattern Analysis and Machine Intelligence 30: 328–341, February 2008.

J. Höhle, M. Höle, 2009, Accuracy assessment of digital elevation models by means of robust statistical methods, ISPRS Journal of Photogrammetry and Remote Sensing, March 2009.

V. Lepetit, M. Moreno-Noguer, P. Fua, 2009, EPnP: An Accurate O(n) Solution to the PnP Problem, International Journal of Computer Vision, February 2009.

A. Martos, 2018, Evaluation Of Optical Distortion Planet Scope 2, Planet Scope Blue And Skysat C, Planet Internal Report, July 2018.

NeoGeographyToolkit Github, StereoPipeline: loading RPC model when stereo with Map-projected Images v2.6.1 #221, https://github.com/NeoGeographyToolkit/StereoPipeline/issues/221#issuecomment-848013231.

OpenCV documentation, Function documentation: calibrateCamera, https://docs.opencv.org/3.4.15/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d.

OpenCV documentation, Function documentation: initUndistortRectifyMap, https://docs.opencv.org/3.4.15/da/d54/group__imgproc__transform.html#ga7dfb72c9cf9780a347fbe3d1c47e5d5a.

Planet Labs, 2021, Planet Imagery Product Specifications, February 2021, https://assets.planet.com/docs/Planet_Combined_Imagery_Product_Specs_letter_screen.pdf.

J. L. Schönberger, J.-M. Frahm, 2016, Structure-from-Motion Revisited, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

G. Slabaugh, R. Schafer, M. Livingston, 2001, Optimal ray intersection for computing 3d points from n-view correspondences, February 2002.

R. Tsai, 1987, A Versatile Camera Calibration Technique For High-Accuracy 3d Machine Vision Metrology Using Off-The-Shelf Tv Cameras And Lenses, IEEE Journal on Robotics and Automation, August 1987.

J. P. de Villiers, F. W. Leuschnerb, R. Geldenhuysb, 2008, Centi-pixel accurate real-time inverse distortion correction, Optomechatronic Technologies 2008: Proc. of SPIE Vol. 7266 726611-1, November 2008.

G. Zhang, X. Yuan, 2006, On RPC Model of Satellite Imagery, Geo-spatial Information Science (Quarterly) Volume 9, 4 December 2006.

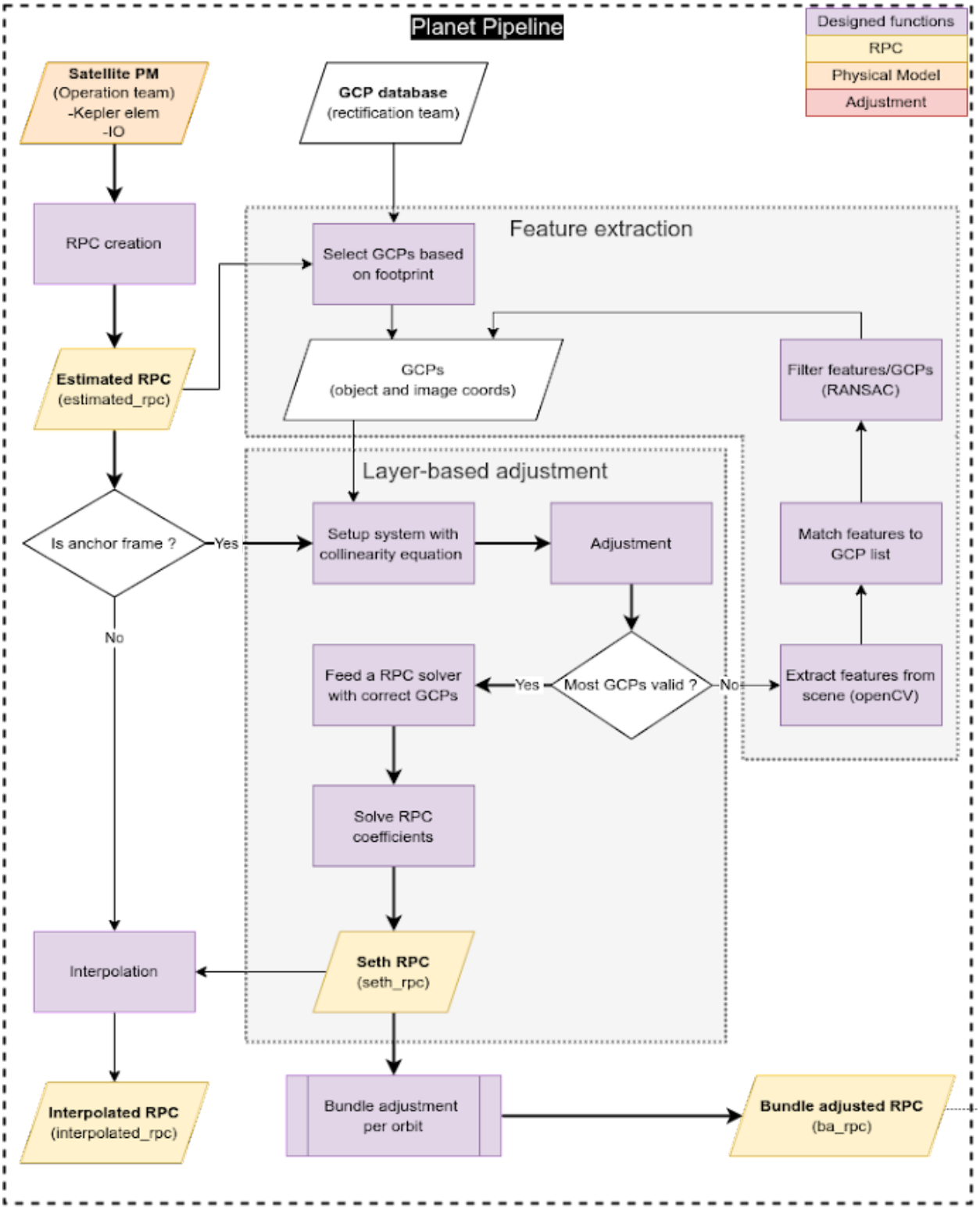
ANNEXES

1. MATHEMATICAL NOTATION

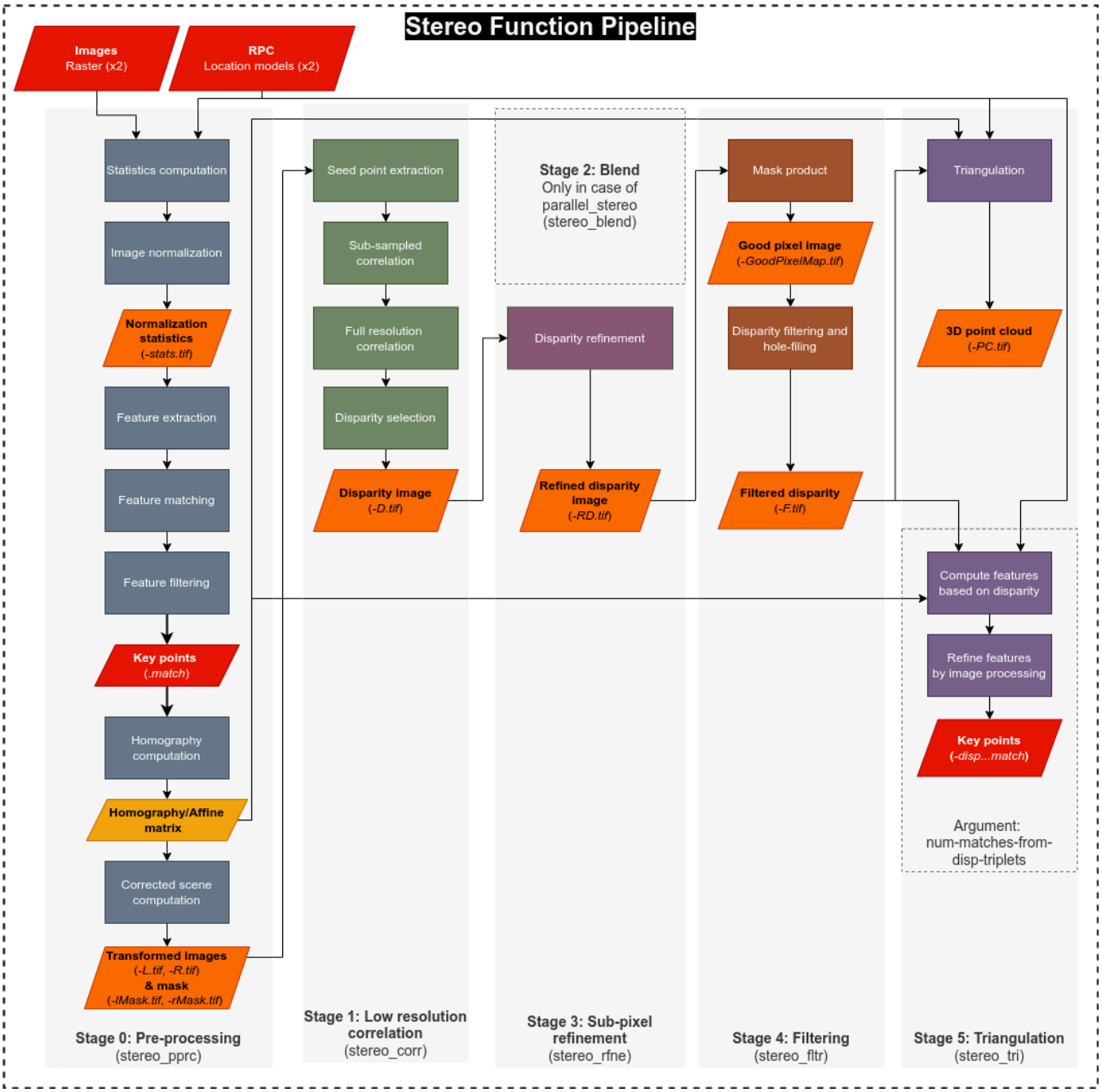
MATHEMATICAL NOTATION IN-USE

Notations	Object side convention	Image side convention
Coordinates components	$\begin{cases}\lambda;\varphi;H \in geographic\\ X;Y;Z \in ECEF\end{cases}$	$x;y$
Normalised components	$\widehat{\lambda};\widehat{\varphi};\widehat{H}$	$\widehat{x};\widehat{y}$
Vector in euclidean space	$\boldsymbol{X}=\begin{pmatrix}X\\Y\\Z\end{pmatrix}$	$\boldsymbol{x}=\begin{pmatrix}x\\y\end{pmatrix}$
Vector in homogeneous space	$\widetilde{\boldsymbol{X}}=\begin{pmatrix}X\\Y\\Z\\\omega\end{pmatrix}$	$\widetilde{\boldsymbol{x}}=\begin{pmatrix}x\\y\\\omega\end{pmatrix}$
RPC function and inverse RPC function	$RPC(\lambda,\varphi,H)$	$RPC^{-1}(x,y,H)$
Geographic to Cartesian conversion and reverse	$G2C(\lambda,\varphi,H)$ $C2G(X,Y,Z)$	
Matrix and inverse matrix	$\boldsymbol{Y};\boldsymbol{A};\boldsymbol{X}=[X_1 \quad X_2 \quad \ldots];\boldsymbol{A}^{-1}$	

2. RPC PROCESS AT PLANET



3. ASP STEREO FUNCTION



4. INVERSE RPC COMPUTATION

Mathematical models (RPCs) compute the object to image transformation (geographic/height to pixel). Its inverse model uses a similar tool to compute the image to object transformation (pixel/height to geographic). Its computation passes by point approximation and returns an independent function from RPC even though we note it $RPC^{-1}(x,y,H)$. It uses a 3rd order rational polynome to ensure a projection accuracy smaller than 1 millimetre.

The inverse computation makes use of normalisation parameters included in RPC (offset, scale). These parameters steady large value computation, as presented by Fraser (2003). They depend on each RPC model in order to yield [-1,1] latitude and longitude, and [-0.2, 0.2] height.

$$\begin{pmatrix}\hat{\lambda}\\ \hat{\varphi}\\ \hat{H}\end{pmatrix}=\begin{cases}\frac{\lambda-o_\lambda}{s_\lambda} & \in[-1,1]\\ \frac{\varphi-o_\varphi}{s_\varphi} & \in[-1,1]\\ \frac{H-o_H}{s_H} & \in[-0.2,0.2]\end{cases}\text{ and }\begin{pmatrix}\hat{x}\\ \hat{y}\end{pmatrix}=\begin{cases}\frac{x-o_x}{s_x} & \in[-1,1]\\ \frac{y-o_y}{s_y} & \in[-1,1]\end{cases}$$

Therefore, we set a slightly broader normalised point grid (9 intermediate values per component) in the object space and project it to the image space using the RPC. We also set the first denominator coefficient to 1 and a least square fits the inverse RPC from these point pairs.

Point pair grid

$$\begin{pmatrix}\hat{x}\\ \hat{y}\end{pmatrix}=RPC_{x,y}(\hat{\lambda},\hat{\varphi},\hat{H})\text{ with }\hat{\lambda}=[-1.1,1.1];\hat{\varphi}=[-1.1,1.1];\hat{H}=[-0.3,0.3]$$

Inverse RPC formula

$$\begin{pmatrix}\hat{\lambda}\\ \hat{\varphi}\end{pmatrix}=RPC^{-1}_{\lambda,\varphi}(\hat{x},\hat{y},\hat{H})=\frac{P_{\lambda,\varphi}(\hat{x},\hat{y},\hat{H})}{Q_{\lambda,\varphi}(\hat{x},\hat{y},\hat{H})}\quad\text{with }\hat{\lambda}=a_0+a_1\,\hat{x}+a_2\,\hat{y}+a_3\,\hat{H}+\ldots-\hat{\lambda}\left(b_1\,\hat{x}+b_2\,\hat{y}+b_3\,\hat{H}+\ldots\right)$$

$$\mathbf{Y}=\begin{bmatrix}\hat{\lambda}_1 & \hat{\varphi}_1 & \hat{\lambda}_2 & \hat{\varphi}_2 & \ldots\end{bmatrix}^T;\ \mathbf{X}=\begin{bmatrix}a_0 & a_1 & \ldots & b_1 & \ldots & c_0 & c_1 & \ldots & d_1 & \ldots\end{bmatrix}^T$$

Least square: $\boldsymbol{Y}=\boldsymbol{AX}+\boldsymbol{\varepsilon}$

$$\mathbf{A}=\begin{bmatrix}1 & \boldsymbol{\Sigma} & -\hat{\lambda}_1\boldsymbol{\Sigma} & & \mathbf{0} & \\ & \mathbf{0} & & 1 & \boldsymbol{\Sigma} & -\hat{\varphi}_1\boldsymbol{\Sigma}\\ 1 & \boldsymbol{\Sigma} & -\hat{\lambda}_2\boldsymbol{\Sigma} & & \mathbf{0} & \\ & \mathbf{0} & & 1 & \boldsymbol{\Sigma} & -\hat{\varphi}_2\boldsymbol{\Sigma}\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\end{bmatrix}\text{ with }\boldsymbol{\Sigma}=\begin{bmatrix}\hat{x} & \hat{y} & \hat{H} & \hat{x}^2 & \ldots\end{bmatrix}$$

Inverse RPC

$$RPC^{-1}()=f(\boldsymbol{X})$$

The accuracy controle is computed with points over the full frame by object to image projection with the RPC and image to object projection with the inverse RPC. We repeat the procedure with 7 RPC files and print out the mean differences in the table beside.

The outcoming error of 1.10^{-9} for latitude and longitude equals to 0.1 mm and thus we can consider our procedure consistent.

Inverse RPC reprojection residuals

Comp:	Long	Lat	Hei
Mean :	-1.60e-09	-1.60e-09	0.00e+00
StdDev:	7.32e-09	7.32e-09	0.00e+00
Mean :	-8.93e-10	-8.93e-10	0.00e+00
StdDev:	7.26e-09	7.26e-09	0.00e+00
Mean :	1.77e-10	1.77e-10	0.00e+00
StdDev:	6.61e-09	6.61e-09	0.00e+00
Mean :	-7.05e-10	-7.05e-10	0.00e+00
StdDev:	6.07e-09	6.07e-09	0.00e+00
Mean :	-4.90e-10	-4.90e-10	0.00e+00
StdDev:	6.01e-09	6.01e-09	0.00e+00
Mean :	-1.30e-09	-1.30e-09	0.00e+00
StdDev:	7.12e-09	7.12e-09	0.00e+00
Mean :	-1.19e-09	-1.19e-09	0.00e+00
StdDev:	7.00e-09	7.00e-09	0.00e+00

5. PERSPECTIVE N POINT (PnP) FROM RPC

We use the PnP algorithm, also called spatial resection (SRS), to extract extrinsic parameters from RPC models. Above all, we notice how such operation is difficult with PS images due to high correlation between all physical parameters. We attempt to circumvent most issues along our two methods providing subpixel solutions but both methods return different solutions. Our methods aim to run geometric processes not related to scene values that makes it independent from image content.

Method 1 with OpenCV function

On one hand, we compute a PnP solution with OpenCV function (*solvePnP*). First of all, we create a point grid over the full image frame using RPC normalised coordinates. That image grid of 11 values per direction from -1 to 1 is projected to the ground using the inverse RPC, as described in the annex 4. We throw that 2D grid to 11 height values from -0.2 to 0.2. It generates a dense point cube with 11^3 points that we convert to cartesian ECEF coordinates then.

$$\begin{pmatrix}\widehat{\lambda}\\ \widehat{\varphi}\end{pmatrix}=RPC_{\lambda,\varphi}^{-1}\left(\widehat{x},\widehat{y},\widehat{H}\right)\ \text{with}\begin{cases}\widehat{x}\ \in[-1.0,1.0]\\ \widehat{y}\ \in[-1.0,1.0]\\ \widehat{H}\ \in[-0.2,0.2]\end{cases}$$

$$\begin{pmatrix}X\\ Y\\ Z\end{pmatrix}=G2C(\lambda,\varphi,H)$$

Method 2 with ASP function

On the other hand, we figured out another way using the ASP *cam_gen* function. It starts with a no distortion RPC computation. An image point grid (11 rows and 11 columns in-between [-1, 1]) projected to several heights (11 height in-between [-0.2, 0.2]) are used to compute the no distortion RPC, named $\widehat{RPC}$. Image coordinates are also corrected by the distortion effect and these pairs fit a no distortion RPC model by least squares.

$$\begin{pmatrix}\widehat{\lambda}\\ \widehat{\varphi}\end{pmatrix}=RPC_{\lambda,\varphi}^{-1}\left(\widehat{x},\widehat{y},\widehat{H}\right)\ \text{with}\begin{cases}\widehat{x}\ \in[-1.0,1.0]\\ \widehat{y}\ \in[-1.0,1.0]\\ \widehat{H}\ \in[-0.2,0.2]\end{cases}$$

$$\boldsymbol{x_u}=\boldsymbol{x}.\left(1+K_1.r^2+K_2.r^4\right)\text{ in [pxl]}$$

$$\boldsymbol{Y}=f(\boldsymbol{x_u});\boldsymbol{A}=f(x_u,y_u,\lambda,\varphi,H);\boldsymbol{Y}=\boldsymbol{AX}+\boldsymbol{\varepsilon}$$

$$\widehat{RPC}=f(\boldsymbol{X})$$

Method 1 (next)

Beside, we pick distortion models up from the satellite ID. As we saw in section 1.7, it is based on the OpenCV formula in pixels. We could have removed the distortion effect to image coordinates using the "*remove*" distortion approximation but we prefer to set the OpenCV formula with converted coefficients since it requires normalised coordinates like the Tsai model in-use during bundle adjustments. We also change the principal point by the distortion centre because the Tsai model does not manage symmetry centre offset. Hence, we let PnP solver overtake that difference. We also build up the camera matrix from manufacturer specification.

$$\boldsymbol{x_d} = \boldsymbol{x_u} + \widehat{\boldsymbol{x}}_{\boldsymbol{u}} \cdot \left(\frac{K_1 \cdot \rho^2}{f^2} \cdot \widehat{r}_u^{\,2} + \frac{K_2 \cdot \rho^4}{f^4} \cdot \widehat{r}_u^{\,4} \right) \text{ with } \mathrm{K_s} \text{ in } [\mathrm{pxl}]$$

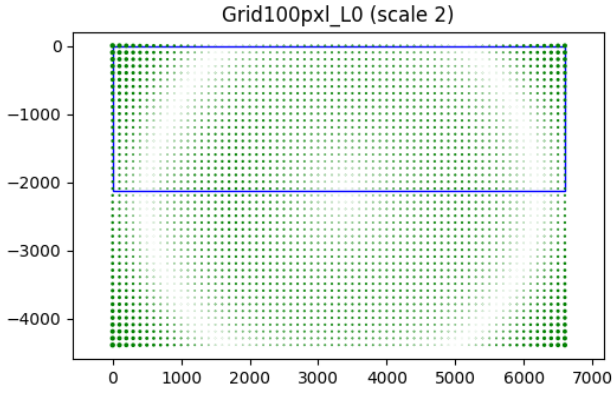
$$\boldsymbol{K}_{3\times3} = \begin{bmatrix} f & 0 & c_u \\ 0 & f & c_v \\ 0 & 0 & \rho \end{bmatrix} \cdot \rho^{-1}$$

Then, we run the EPnP algorithm from Lepetit *et al.* (2008) using virtual intermediate points to solve extrinsic parameters. That is the most successful algorithm under our picky conditions. The camera orientation expressed as rotation vector is converted into a rotation matrix by matrix exponential of the skew-symmetric matrix of the returned vector. The camera centre is transformed from the camera system to the ECEF system by product with the transposed rotation matrix. There is no need to refine the EPnP solution since it is already embedded in the Lepetit (2008) algorithm.

Method 2 (next)

Then, we run the *cam_gen* function from ASP which converts $\widehat{RPC}$ into PM. We force a dense point grid over L0 shape and we include the refinement argument. That function creates 3D vectors for each image point (by approximating RPC) and intersects them with the user DEM, SRTM in our case. These point pairs fit a PnP solver and return an extrinsic model. Intrinsic ones are fixed by user arguments. A small height range in spatial resection leads to a weak solution which is improved during the image residual refinement.

A similar process with an initial RPC instead of $\widehat{RPC}$ leads to a worse solution due to distortion effects. During the refinement, a least square balances image errors around zeros while the embedded distortion should have a constant sign. The following figure presents the image residual in such a case. Inner residuals are negative while outer ones are positive. That effect can be reduced with a circular point grid in the central part but that grid does not surround the full frame and that shortcut does not return sub-pixel accuracy.



Hence, both methods create sub-pixel physical models from RPC. We keep in mind that those methods provide better results with large point ranges but distant points may be beyond the validity range of RPC or distortion. That is also the reason why we build point grids in image space instead of geographic range. To conclude, There are a few hundreds of metres difference at the camera centre

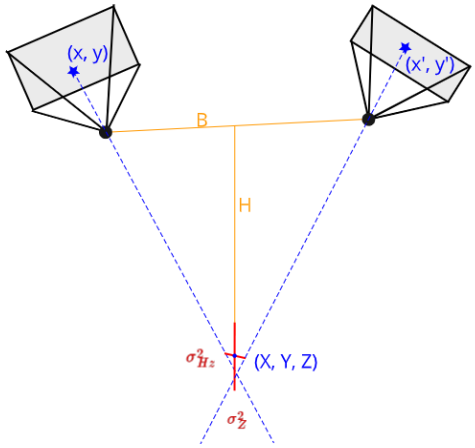
between both PMs and the projection of random points returns a sub-pixel accuracy. Although, it happens that they different by more than 1 km depending on the method. We suspect that adjusted coefficients in Planet RPCs model without physical meaning which are not fittable. We prefer the OpenCV method because it uses a large 3D point cube, does not depend on external DEM and returns key points closer to RPC projection at the initial stage of bundle adjustments.

6. WEIGHTED RASTERIZATION

At the last stage, our MVS method merges the height redundancy using the point quality redundancy into a raster file at the original scene GSD. A weighted average returns the merged elevation and the Standard Error of weighted Mean (SEM) value. In the following, we present the weight computation from input information and the height average derivation as well as the SEM derivation. Due to some limitations in the PDAL rasterizing function, we also write out command line expressions with function parameters denoted in **bold**.

The dense matching process provides point coordinates with their elevation written Z as well as their intersection error. This error is the distance in metres between 2 intersecting rays as shown in the following figure, stored in millimetres in *las* files. We assume these values to be uncorrelated horizontal variance σ^2_{Hz} of points. The incidence angle α computed from the B/H ratio is also stored. Moreover, the LiDAR morphological filters classify points from cloud morphology as noise or not (binary state: 0=not noise, 7=noise), named c.

From intersection variance to height variance



Firstly, we compute a point weight by input accuracy combination. The horizontal variance of a point is transformed into vertical variance using the B/H ratio. The incidence angle can be assumed small (<15°) and thus the B/H ratio is equivalent to the incidence angle in radian. We allow such simplification since the angle value is already rounded to fit an integer field. The point weight is the inverse elevation variance that we shift by 1 m to avoid division by zero and we scale it by 1000 to record it as integer value. Hence, the link between the weight and the point variance remains through a scale factor and an offset. The point class discards noise point by product to its binary expression (0: noise, 1: valid).

$$\frac{B}{H} = 2.tan(\frac{a}{2}) \simeq \alpha[\boldsymbol{rad}] \qquad \sigma_Z^2[\boldsymbol{m}] = \frac{\sigma_{Hz}^2[\boldsymbol{m}]}{\alpha[\boldsymbol{rad}]}$$

$$w = \frac{1}{\sigma_Z^2} = \frac{\alpha[\boldsymbol{deg}]}{\sigma_{Hz}^2[\boldsymbol{mm}]}.\frac{\pi \times 1000}{180} \qquad \sigma_Z^2[\boldsymbol{m}] = \frac{1000}{w}$$

$$w = \frac{1}{\sigma_Z^2 + offset}.scale.c = \frac{\alpha}{\sigma_{Hz}^2 + 1000}.\frac{\pi \times 1000^2}{180}.c = \frac{\boldsymbol{ScanAngleRank}}{\boldsymbol{Intensity} + 1000}.\frac{\pi \times 1000^2}{180}.c$$

Then, merged height is provided by weighted average. We decompose it in accordance with PDAL rasterization options.

$$\overline{Z} = \frac{\sum_i^N (w_i Z_i)}{\sum_i^N (w_i)} = \frac{\sum_i^N (w_i.Z_i)}{N}.\left(\frac{\sum_i^N (w_i)}{N}\right)^{-1} = \frac{\boldsymbol{RasterMean}(w_i.Z_i)}{\boldsymbol{RasterMean}(w_i)}$$

The variance of weighted mean per cell is given by the error propagation. After simplification, the lighter formula can be decomposed in accordance with PDAL options. We derive the SEM after scaling and offset correction.

$$\overline{\sigma_Z^2} = \frac{\sum_i^N (w_i^2.\sigma_Z^2)}{\left(\sum (w_i)\right)^2} = \frac{\sum_i^N (w_i^2.\sigma_Z^2)}{N}.\left(\frac{\sum_i^N (w_i)}{N}\right)^{-2} * \frac{1}{N} = \frac{\sum_i^N (\sigma_Z^2.\sigma_{\overline{Z}}^4)}{N}.\left(\frac{\sum_i^N (\sigma_{\overline{Z}}^2)}{N}\right)^{-2}.\frac{1}{N} = \left(\frac{\sum_i^N (\sigma_{\overline{Z}}^2)}{N}\right)^{-1}.\frac{1}{N}$$

$$= \left(\boldsymbol{RasterMean}(w_i).\boldsymbol{RasterCount}()\right)^{-1}$$

$$\overline{\sigma_Z} = \sqrt{\overline{\sigma_Z^2}.1000 - 1} = \sqrt{\frac{1000}{\boldsymbol{RasterMean}(w_i).\boldsymbol{RasterCount}()} - 1}$$

Finally, we create the final product with the merged height in 1st band, the height accuracy in the 2nd band and the number of included points in the 3rd band. The recorded accuracy is the sum of the SEM and the point standard deviation computed by the rasterization tool.

$$\overline{\sigma_H} = \overline{\sigma_Z} + \sigma_P = \sqrt{\frac{1000}{\boldsymbol{RasterMean}(w_i).\boldsymbol{RasterCount}()} - 1} + \boldsymbol{RasterStdev}()$$

Table 8: final product description

Digital Surface Model (DSM)	Band 1: float 32 bits, “Height” (merged height $\overline{Z}$) Band 2: float 32 bits, “Accuracy” (height accuracy $\overline{\sigma_H}$) Band 3: float 32 bits, “PtCount” (point count N)