



|   |                    |                   |
|---|--------------------|-------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: 10/27/ 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A       |

# NEON ALGORITHM THEORETICAL BASIS DOCUMENT (ATBD) AOP DIGITAL CAMERA IMAGE ORTHORECTIFICATION

| PREPARED BY   | ORGANIZATION | DATE       |
|---|--------------|------------|
| William Gallery, Tristan Goulden, Nathan Leisso, Keith Krause | AOP          | 03/02/2015 |
|   |              |            |

| APPROVALS    | ORGANIZATION | APPROVAL DATE |
|--------------|--------------|---------------|
| Hanne Buur   | PSE          | 10/06/2015    |
| Dave Tazik   | SCI          | 10/02/2015    |
| Andrea Thorp | SCI          | 10/23/2015    |
|              |              |               |

| RELEASED BY  | ORGANIZATION | RELEASE DATE |
|--------------|--------------|--------------|
| Judy Salazar | CM           | 10/27/2015   |

See configuration management system for approval history.

© 2015 NEON Inc. All rights reserved.

The National Ecological Observatory Network is a project solely funded by the National Science Foundation and managed under cooperative agreement by NEON, Inc. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.



|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |

## Change Record

| <b>REVISION</b> | <b>DATE</b> | <b>ECO #</b> | <b>DESCRIPTION OF CHANGE</b> |
|-----------------|-------------|--------------|------------------------------|
| A               | 10/27/2015  | ECO-02762    | Initial release              |
|                 |             |              |                              |
|                 |             |              |                              |

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

**TABLE OF CONTENTS**

**TABLE OF CONTENTS** ..... **III**

**LIST OF TABLES AND FIGURES** ..... **V**

**1 DESCRIPTION** ..... **1**

    1.1 Purpose ..... 1

    1.2 Scope ..... 1

**2 RELATED DOCUMENTS AND ACRONYMS** ..... **2**

    2.1 Applicable documents ..... 2

    2.2 Reference documents ..... 2

    2.3 External Documents ..... 2

**3 DATA PRODUCT DESCRIPTION** ..... **3**

    3.1 Variables reported ..... 3

    3.2 Input Dependencies ..... 3

    3.3 Product instances ..... 3

    3.4 Temporal resolution and extent ..... 3

    3.5 Spatial resolution and extent ..... 4

**4 SCIENTIFIC CONTEXT** ..... **5**

    4.1 Theory of measurement/observation ..... 5

        4.1.1 The D8900 digital camera ..... 8

    4.2 Theory of algorithm ..... 8

        4.2.1 Correcting the raw images for color balance and exposure ..... 9

        4.2.2 Creating a camera and alignment model ..... 10

        4.2.3 Complete camera model ..... 12

        4.2.4 Calculating the camera parameters ..... 13

        4.2.5 Orthorectifying the Corrected Images ..... 20

**5 ALGORITHM IMPLEMENTATION** ..... **23**

    5.1 Color balance and exposure correction ..... 25

        5.1.1 Requirements ..... 25

        5.1.2 Processing steps ..... 25

    5.2 Camera model creation ..... 26

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

|                   |   |           |
|-------------------|---|-----------|
| 5.2.1             | Requirements .....  | 26        |
| 5.2.2             | Processing steps .....                                    | 26        |
| 5.3               | Camera image orthorectification .....                     | 28        |
| 5.3.1             | Requirements .....  | 28        |
| 5.3.2             | Processing steps .....                                    | 28        |
| <b>6</b>          | <b>UNCERTAINTY.....</b>                                   | <b>31</b> |
| 6.1               | Analysis of uncertainty .....                             | 31        |
| 6.2               | Reported uncertainty .....                                | 31        |
| <b>7</b>          | <b>VALIDATION AND VERIFICATION .....</b>                  | <b>32</b> |
| 7.1               | Data product validation .....                             | 32        |
| 7.2               | Data product verification.....                            | 32        |
| <b>8</b>          | <b>FUTURE MODIFICATIONS AND PLANS.....</b>                | <b>33</b> |
| 8.1               | Establish accurately measured ground control points ..... | 33        |
| 8.2               | Move from proprietary to non-proprietary software .....   | 33        |
| 8.3               | Refine the resampling procedure.....                      | 33        |
| 8.4               | Mosaic and tile the images.....                           | 33        |
| <b>9</b>          | <b>CHANGE LOG .....</b>                                   | <b>34</b> |
| <b>10</b>         | <b>REFERENCES .....</b>                                   | <b>35</b> |
| <b>APPENDIX A</b> | <b>MATHEMATICS OF GEOLOCATION .....</b>                   | <b>36</b> |
| A.1               | Geolocation for the shooting-down method .....            | 36        |
| A.2               | Shooting up .....   | 44        |
| A.3               | Shooting down vs shooting up: tradeoffs .....             | 46        |
| A.4               | Bibliography .....  | 47        |
| <b>APPENDIX B</b> | <b>BROWN’S CAMERA DISTORTION MODEL .....</b>              | <b>48</b> |

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## LIST OF TABLES AND FIGURES

|  |    |
|--|----|
| Table 1. Reported Variables for Data Product NEON.dom.site.DP1.30008.....  | 3  |
| Table 2. Input Dependencies .....  | 3  |
| Table 3. Spatial resolution and extent of the raw digital camera images .....  | 4  |
| Table 4. Specifications of the D8900 Camera .....  | 8  |
| Table 5 Camera model parameters .....  | 13 |
| Table 6. Color balance and exposure correction: required inputs and parameters.....  | 25 |
| Table 7. Camera model creation: required inputs and parameters .....   | 26 |
| Table 8. Template input control file for the program run_neon_aig_dms_cal_GCP_and_tp.pro .....   | 27 |
| Table 9. Camera image orthorectification: required inputs and parameters.....  | 28 |
| Table 10 Template input control file for the program run_neon_aig_dms_ortho.pro .....  | 29 |
| Table B-1. List of camera model parameters.....  | 48 |
| Table B-2. Camera model parameters for three models .....  | 49 |
| <br>   |    |
| Figure 1. Sketch of the image observation .....  | 6  |
| Figure 2. Representative camera image without and with orthorectification .....  | 7  |
| Figure 3. Top level processing flow diagram.....   | 9  |
| Figure 4. Geometry of a pinhole camera .....   | 11 |
| Figure 5. Outline of the flight tracks over Fruita, CO, 07/17/2013 .....   | 14 |
| Figure 6. LiDAR intensity image and overlapping camera image showing a matching object .....   | 15 |
| Figure 7. LiDAR image, orthorectified camera image and raw camera image centered on the<br>same object. ....                                 | 16 |
| Figure 8. Two overlapping digital images showing a tie point:.....   | 17 |
| Figure 9. Location error for camera model NEON_DiMAC_8900_B_001.....   | 19 |
| Figure 10. Sketch of camera above the ground showing the detector grid and the image grid on<br>the ground .....                             | 21 |
| Figure 11. Sketch of a 50 by 50 image grid on the ground showing the locations where the<br>camera edge pixels (X) intersect the ground..... | 21 |
| Figure 12. Digital camera orthorectification top-level flow chart .....  | 23 |
| Figure 13. Camera model creation flow chart .....  | 24 |
| Figure 14. Camera image orthorectification flow chart.....   | 24 |

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## 1 DESCRIPTION

### 1.1 Purpose

This document details the algorithms used for creating NEON Level 1 data product NEON.*dom.sit*.DP1.30008.001.001.001.001 Digital Camera Orthorectified Images<sup>1</sup>. It includes a detailed discussion of measurement theory and implementation, appropriate theoretical background, data product provenance, quality assurance and control methods used, approximations and/or assumptions made, and a detailed exposition of uncertainty resulting in a cumulative reported uncertainty for this product.

### 1.2 Scope

This document describes the theoretical background and entire algorithmic process for creating NEON Level 1 data product NEON.*dom.sit*.DP1.30008.001.001.001.001 Digital Camera Orthorectified Images from input data

---

<sup>1</sup> In the product name, *dom* and *site* refer to the NEON domain and site respectively. See AD[02].

## 2 RELATED DOCUMENTS AND ACRONYMS

### 2.1 Applicable documents

|        |                 |   |
|--------|-----------------|---|
| AD[01] | NEON.DOC.000001 | NEON Observatory Design                                 |
| AD[02] | NEON.DOC.005003 | NEON Scientific Data Products Catalog                   |
| AD[03] | NEON.DOC.002652 | NEON Level 1, Level 2 and Level 3 Data Products Catalog |
| AD[04] | NEON.DOC.005005 | NEON Level 0 Data Product Catalog                       |
| AD[05] | NEON.DOC.000254 | NEON Data Products Naming Convention                    |

### 2.2 Reference documents

|        |                      |  |
|--------|----------------------|--|
| RD[01] | NEON.DOC.000008      | NEON Acronym List                                      |
| RD[02] | NEON.DOC.000243      | NEON Glossary of Terms                                 |
| RD[03] | NEON.DOC.001541      | NEON AOP Payload Definition                            |
| RD[04] | NEON.DOC.001292      | NEON L0 to L1 Discrete Return LiDAR ATBD               |
| RD[05] | NEON.DOC.001207      | NEON Imaging Spectrometer Geolocation ATBD             |
| RD[06] | NEON AOP Tech Report | NEON_D8900 Delivery Report EH021537.pdf                |
| RD[07] | NEON DOC Tech Report | NEON AOP Procedure for Processing the GPS and IMU Data |
| RD[08] | NEON DOC Tech Report | NEON AOP Operations Overview (TBD)                     |
|        |                      |  |

### 2.3 External Documents

|        |  |
|--------|--|
| ED[01] | <a href="http://cddis.gsfc.nasa.gov/egm96/egm96.html">The NASA GSFC and NIMA Joint Geopotential Model: http://cddis.gsfc.nasa.gov/egm96/egm96.html</a>   |
| ED[02] | <a href="http://www.phaseone.com/en/Imaging-Software/Capture-One-Pro-7.aspx">Capture One Pro 7 - photo editing software: http://www.phaseone.com/en/Imaging-Software/Capture-One-Pro-7.aspx</a>  |
| ED[03] | <a href="http://help.phaseone.com/en/CO7/Output/File-formats/Capture-One-and-RAW.aspx#item3">II Q format: http://help.phaseone.com/en/CO7/Output/File-formats/Capture-One-and-RAW.aspx#item3</a> |
| ED[04] | <a href="http://www.phaseone.com/en/FooterMenu/AboutUs.aspx">Phase One: http://www.phaseone.com/en/FooterMenu/AboutUs.aspx</a>   |
| ED[05] | <a href="http://rawtherapee.com/blog/features">Raw Therapee: http://rawtherapee.com/blog/features</a>  |
| ED[06] | <a href="http://www.chrisevansdev.com/computer-vision-opensurf.html">OpenSURF: http://www.chrisevansdev.com/computer-vision-opensurf.html</a>  |
| ED[07] | <a href="http://tortoisesvn.net/">Subversion (SVN): see http://tortoisesvn.net/</a> for an MS Windows svn client and further links   |

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

### 3 DATA PRODUCT DESCRIPTION

#### 3.1 Variables reported

The data product produced by this processing is listed in Table 1. See AD[05] for an explanation of the data products naming convention.

**Table 1. Reported Variables for Data Product NEON.dom.site.DP1.30008**

| Variable                             | Units      | Product Number                          |
|--------------------------------------|------------|---|
| Digital Camera Orthorectified Images | (unitless) | NEON.dom.site.DP1.30008.001.001.001.001 |

#### 3.2 Input Dependencies

The required inputs for this processing are listed in Table 2.

**Table 2. Input Dependencies**

| Input            | Reference | Description   |
|------------------|-----------|---|
| Camera images    | L0 data   | Raw camera images   |
| DiMAC log file   | L0 data   | GPS time marks and file names of raw images                             |
| Sbet file        | TBD       | Trajectory file (sbet = smoothed best estimate of trajectory)           |
| DSM file         | TBD       | Digital surface map (DSM) from the discrete LiDAR <sup>2</sup>          |
| LiDAR images     | TBD       | Orthorectified LiDAR “intensity images” used to create the camera model |
| EGM96 geoid file | ED[01]    | Earth’s gravitational model   |

#### 3.3 Product instances

The product described by this ATBD is NEON.dom.site.DP1.30008.001.001.001.001, Digital Camera Orthorectified Images.

#### 3.4 Temporal resolution and extent

Reference RD[08] describes the operation of the AOP instruments, including the digital camera. A flight consists of a number of straight flightlines arranged so that the fields-of-view of neighboring flight lines overlap, typically between 30 to 50 percent. The camera takes images every 0.5 seconds along a flight line. The flight parameters are set so that the camera images overlap by 60% along the direction of travel.

<sup>2</sup> NEON AOP has adopted the following convention: DSM refers to ‘digital surface model’, which includes the top of buildings, trees, etc and is derived from the LiDAR first return. DTM refers to ‘digital terrain model’ and is the elevation of the bare earth with structures and trees removed. DEM refers to ‘digital elevation model’ and can mean either DTM or DSM.



Orthorectification of the camera images requires a digital surface model (DSM) which is derived from the LiDAR. Some LiDAR data may not be included in the DSM and therefore some camera images may extend beyond the DSM and cannot be orthorectified.

### 3.5 Spatial resolution and extent

The camera image size is 8,984 x 6,732 pixels (see Table 4.) The spatial resolution and extent of the raw images depends upon flight altitude above ground level (AGL). The AOP normally flies at one of two nominal altitudes: 1000 m or 1500 m depending on the terrain. Table 3 summarizes spatial resolution and extent for different flight altitudes (500 m is included for information only.)

**Table 3. Spatial resolution and extent of the raw digital camera images**

|                      | Altitude (AGL) |            |             |
|----------------------|----------------|------------|-------------|
|                      | 500 m          | 1000 m     | 1500 m      |
| Pixel resolution (m) | 0.043          | 0.086      | 0.129       |
| Total footprint (m)  | 385 by 289     | 770 by 577 | 1155 by 866 |

The orthorectified images are resampled to 0.25-meter resolution for delivery as the final data product. See Section 4.2.5.2 for the motivation for down sampling.

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## 4 SCIENTIFIC CONTEXT

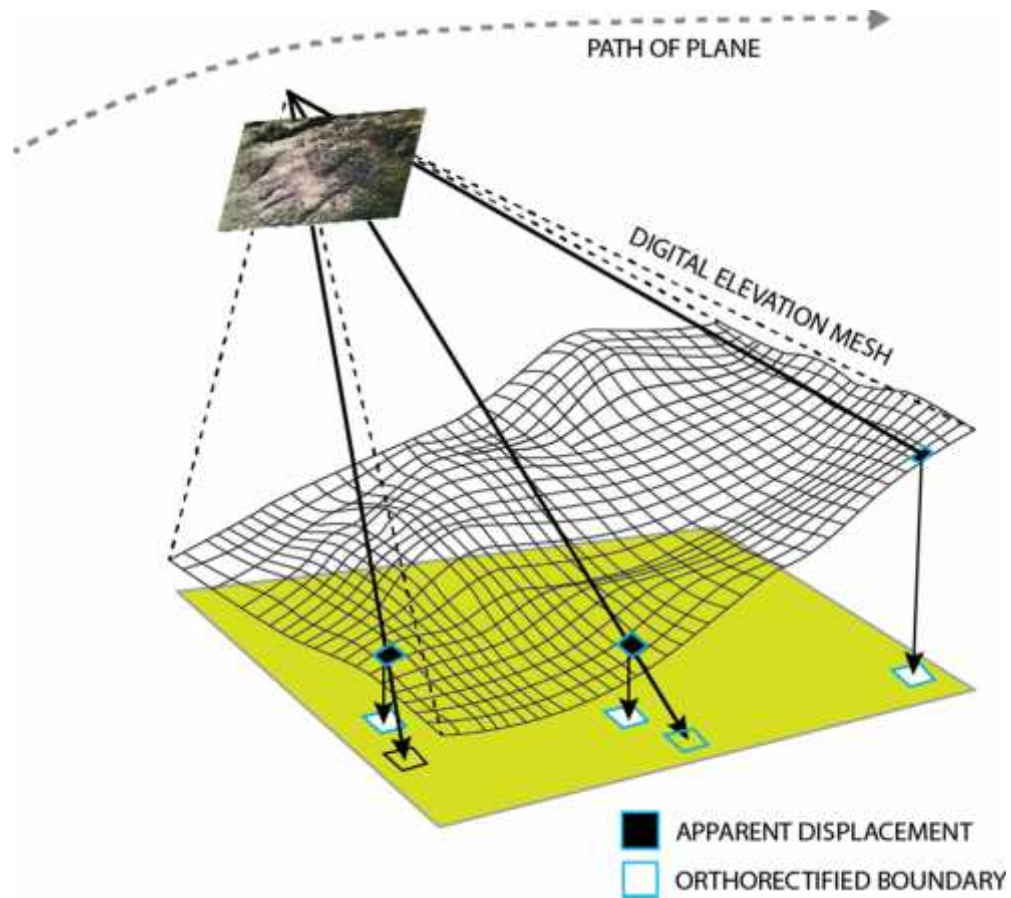
The digital camera is part of a suite of instruments on the NEON Airborne Observation Platform (AOP) that also includes a full-waveform, small-footprint LiDAR system and the NEON Imaging Spectrometer. The AOP is an airborne system developed for monitoring terrestrial ecosystems and is typically integrated onto a Twin Otter aircraft that flies at approximately 100 knots airspeed and at a nominal altitude of 1000 m AGL. The AOP instrument suite provides regional scale observations of vegetation structure, biochemical and biophysical properties of vegetation, and ecosystem responses to changes in land use, climate and the movement of invasive species. The complete AOP instrument suite is described in RD[03].

Orthorectification is the process of remapping the pixels in an image from the native projection to a regular geographic projection while removing distortions introduced by land surface topography, imperfections in the camera lens, and tilt of the camera caused by aircraft attitude variations. In the orthorectification process, the digital imagery is remapped to the same geographic projection as the LiDAR and imaging spectrometer data that is acquired simultaneously. The resulting images will share the same map projection grid space as the orthorectified spectrometer and LiDAR imagery. Since the digital camera imagery is acquired at higher spatial resolution than the imaging spectrometer data, it can aid in identifying features in the spectrometer images including manmade features (e.g., roads, fence lines, and buildings) that are indicative of land-use change.

### 4.1 Theory of measurement/observation

The digital camera takes images from an altitude of from 500 to 1500 m AGL and is oriented to capture vertical images. Images are captured in a snapshot mode, and assuming a nominal flight altitude of 1000 m AGL, each image encompasses an area of 770 m cross-track by 577 m along-track (44 deg x 33 deg). Images are collected at a rate designed to 50 percent overlap between adjacent images. At an altitude of 1000 meters and a ground speed of 100 knots, an image is collected about every 5.7 seconds.

Within a single image acquisition, the topography may vary and the actual view angle of the image center will deviate from nadir due to the roll, pitch and yaw of the aircraft. As a result, the native projection of the image will not correspond to a regular grid on the ground. Figure 1 shows the geometry of the observation. The aircraft flies over terrain of varying elevation. Orthorectification transforms the image in the original projection to what would be seen by an observer viewing the scene from vertically above at a great distance.



**Figure 1. Sketch of the image observation**

Figure 2 shows a sample image before and after orthorectification. The orthorectified image is created on a regular geographic grid with the x- and y-axes aligned east and north respectively. The grid size of the orthorectified image is approximately the same as the average pixel size of the original image, in this case, 8 cm. Each pixel in the original raw image is mapped to the appropriate location on the geographic grid. The black areas are regions in geographic space not covered by the original image. Note the curved edge on the upper right side of the orthorectified image: the curvature is caused by a steep elevation gradient in this area.

|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |



a.



b.

**Figure 2. Representative camera image without and with orthorectification: a. the raw image, b. the orthorectified image**

#### 4.1.1 The D8900 digital camera

The [D8900](#) is a commercial off-the-shelf digital color camera sold by [Optech, Inc.](#), of Ontario, Canada. Table 4 lists the camera specifications (from [D8900](#).) For this application, the camera uses a 70mm focal length lens, giving a cross-track field-of-view of 42.1 deg. For the camera delivery report, see RD[06]

**Table 4. Specifications of the D8900 Camera**

| Parameter                            | Specification   |
|--------------------------------------|---|
| <b>Camera Head</b>                   |   |
| Sensor type                          | 60 Mpix full frame CCD, RGB   |
| Sensor format (H x V)                | 8,984 x 6,732 pixels  |
| Pixel size                           | 6 $\mu\text{m}$ x 6 $\mu\text{m}$   |
| Frame rate                           | 1 frame/2 seconds maximum   |
| Forward motion compensation (FMC)    | Electro-mechanical, driven by piezo technology (patented)   |
| Shutter                              | Electro-mechanical iris mechanism 1/125 to 1/500 sec.<br>f-stops: 5.6, 8, 11, 16  |
| Lenses                               | 50 mm/70 mm/120 mm/210 mm   |
| Filter                               | Color and near-infrared removable filters   |
| Dimensions (H x W x D)               | 200 x 150 x 120 mm (70 mm lens)   |
| Weight                               | ~4.5 kg (70 mm lens)  |
| <b>Controller Unit</b>               |   |
| Computer                             | Mini-ITX RoHS-compliant small-form-factor embedded computers with AMD TurionTM 64 X2 CPU, 4 GB RAM, 4 GB flash disk local storage, IEEE 1394 Firewire interface |
| Removable storage unit               | ~500 GB solid state drives, 8,000 images  |
| Power consumption                    | ~8 A, 168 W   |
| Dimensions                           | 2U full rack; 88 x 448 x 493 mm   |
| Weight                               | ~15 kg  |
| <b>Image Pre-Processing Software</b> |   |
| CaptureOne                           | Radiometric control and format conversion, TIFF or JPEG   |
| Image output                         | 8,984 x 6,732 pixels, 8 or 16 bits per channel (180 MB or 360 MB per image)   |

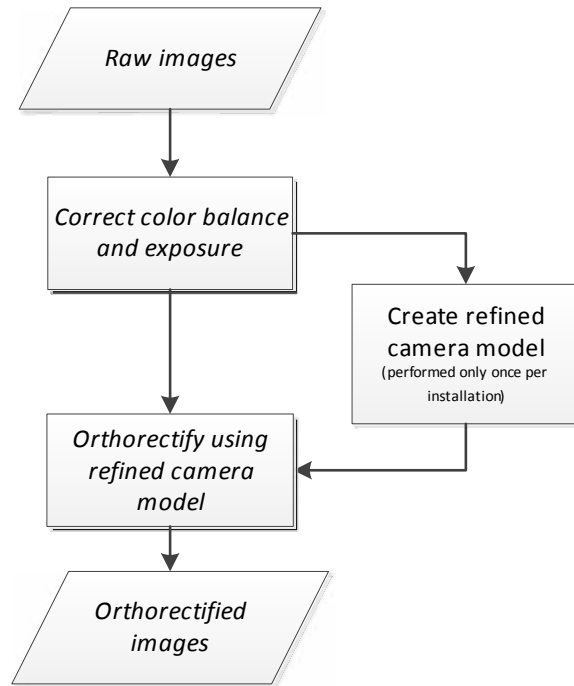
The images collected by the D8900 have a '.tiff' extension but are actually in [II Q format](#) (ED[03]), a proprietary format from [Phase One](#) (ED[04]). They cannot be read properly by most software: many image viewers can show only a 540 by 480 thumbnail. They can be displayed and processed by [Camera One](#) (ED[02]) and by [Raw Therapee](#) (ED[05]).

#### 4.2 Theory of algorithm

The complete processing sequence involves three steps:

1. Correcting the raw images for color balance and exposure
2. Creating a camera and alignment model (this step is performed only once per installation)
3. Orthorectifying the corrected images

The top level processing flow is shown in Figure 3.



**Figure 3. Top level processing flow diagram**

These steps are covered separately in the following sections.

Before discussing these steps, the reader is referred to **Appendix A. Mathematics of geolocation**. This section introduces the three frames of reference involved in orthorectifying the camera images and describes the mathematical operations used in geolocation.

#### 4.2.1 Correcting the raw images for color balance and exposure

The goal of this processing step is to:

1. Correct individual images for optical vignetting, color balance and exposure, and
2. Allow the creation of a seamless mosaic where all of the images match in terms of color balance and brightness.

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

Most camera systems suffer from lens illumination falloff due to optical vignetting. This causes the corners of the raw image to be darker than the center of the image. A lens correction needs to be applied so that a uniform target shows up with a uniform brightness across the entire processed camera image.

The D8900 digital camera uses a PhaseOne P65+ digital back as the focal plane. The digital back records 12-bit raw data for each of the three color channels or bands. No radiometric calibration has been applied to these raw images so the histograms in the red, green, and blue channels will not necessarily match each other. This means that a spectrally uniform gray target will have different digital number (DN) values for each band in the raw image and will show up with a color hue. The raw camera images need to be adjusted for this incorrect color balance typically using a “white balance” procedure where a spectrally uniform white or gray pixel is selected and used to adjust the three-color bands in order for that pixel DN value to be the same in all three bands. If a good (i.e., spectrally uniform) pixel is selected during this procedure, the overall color hue of the image will be eliminated.

The lighting conditions and brightness of targets on the ground seen by the camera will vary throughout the flight due to changes in the sun elevation and azimuth, and to atmospheric conditions such as clouds and cloud shadows. The D8900 has several different aperture f-stops that can be set to control the exposure of the raw image so that it does not underexpose or overexpose images. The airborne sensor operators (ASOs) routinely monitor image histograms and determine if the f-stop needs to be adjusted to change the exposure due to changes in the illumination or differences in ground target types (for example a dark green forest versus a cleared area of bright exposed soil). Nonetheless, some variation in exposure across the different images remains, an exposure correction procedure must be applied to the camera images during processing to correct for the differences in brightness (contrast could also be adjusted but is not currently modified during image processing). Currently, these adjustments are performed using the program “[Capture One 7](#)” (ED[02]) by [Phase One](#) (ED[04]). The details of this process are given in Section 5.1.

#### 4.2.2 Creating a camera and alignment model

Orthorectification of the camera images requires the accurate knowledge of line-of-sight (LOS) of each image pixel relative to the camera reference frame (the camera model) and of the boresight of the camera relative to the position and orientation system (POS) reference frame (the alignment model).

The camera model is a mathematical function that maps the pixels in the sensor plane to the corresponding location in object space. The camera model can be divided into two parts:

1. Geometric: which implements the “collinearity equation” of geometric optics,
2. Distortion: which corrects for distortions caused by the camera optics

The camera geometric model is that of a pinhole camera. The camera distortion model used here is known as “Brown’s model (Brown 1965). See Appendix B in this document for a description of this model. The alignment model describes the displacement and orientation of the camera focal plane relative to the POS reference frame.

#### 4.2.2.1 Collinearity Equation

For this part of the discussion, we can assume that the camera is modeled as a pinhole camera (Figure 4). Let the s-frame be a coordinate system fixed in the sensor plane as shown, with the z-axis perpendicular to the sensor plane. A point  $P$ , with coordinates  $x_p, y_p, z_p$ , is imaged onto the sensor plane at  $P'$  with coordinates  $x, y, (z = 0)$ . The aperture, or focal point ( $O$ ), has the coordinates  $x_0, y_0, z_0$  ( $z_0$  is also the focal length  $f$  of the camera). The collinearity equation (Schenk 2014) states that:

$$x - x_0 = -\frac{z_0}{z_p - z_0}(x_p - x_0)$$

$$y - y_0 = -\frac{z_0}{z_p - z_0}(y_p - y_0)$$

eq. 1

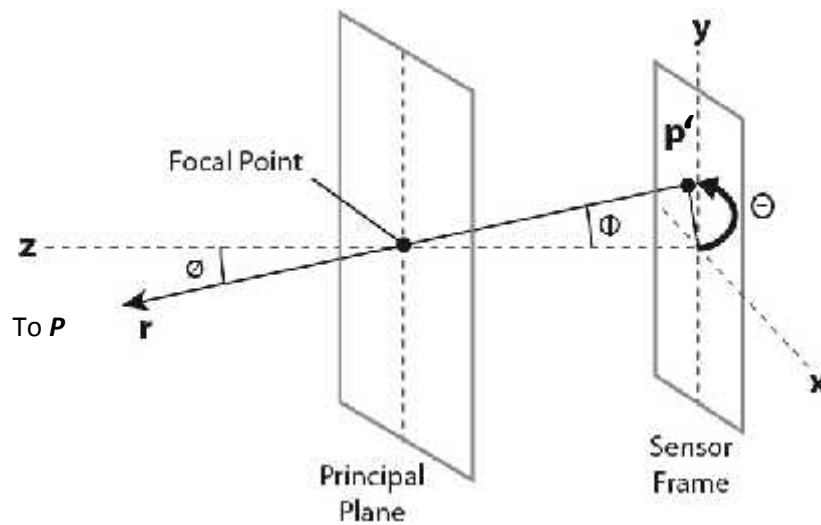


Figure 4. Geometry of a pinhole camera

The principal point offsets  $x_0, y_0$  are nominally 0,0 and the focal length  $f = z_0$  is obtained from the camera manufacturer. However, the camera model fitting process generates refined values for these parameters.



#### 4.2.2.2 Distortion model

Real cameras are subject to distortions not accounted for in the simple pinhole model of the previous section. In general, these distortions include radial distortion, tangential distortion, decentering distortion and others. These distortions can be corrected using a “camera distortion model” which relates the undistorted pixel locations  $x_u, y_u$  as predicted by the collinear equations (eq. 1) to the distorted pixel locations  $x_d, y_d$ . Here we use Brown’s camera model, as described in Appendix B. The results of this model are given in eq. B-1 repeated here as eq. 2:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} x_u \\ y_u \end{bmatrix} \left( 1 + K_1 r^2 + K_2 r^4 + K_3 r^6 \right) + \begin{bmatrix} \left( P_1 (r^2 + 2x_u^2) + 2P_2 x_u y_u \right) \\ \left( P_2 (r^2 + 2y_u^2) + 2P_1 x_u y_u \right) \end{bmatrix} \left( 1 + P_3 r^2 + P_4 r^4 \right) \quad \text{eq. 2}$$

However, experience with the D8900 camera has shown that the distortion corrections provided by Brown’s camera model are insignificant and can be ignored (see Section 6.1.) The capability for using Brown’s model has been retained in the processing software and can be applied in the future if needed.

#### 4.2.2.3 Alignment model

Alignment refers to the location and orientation of the camera sensor frame ( $s$ -frame) relative to the aircraft body frame ( $b$ -frame.) See Appendix A.1 and A.2 for a discussion of these parameters. The results of the alignment model populate the lever-arm ( $\mathbf{l}$ ) vector in the  $b$ -frame, and the boresight misalignment values ( $\beta$ ), which angularly relate the  $s$ -frame to the  $b$ -frame and populate a rotation matrix  $\mathbf{R}_s^b$ . These components are written in matrix notation as

$$\mathbf{l}_b = (x_l, y_l, z_l) \quad \text{eq. 3}$$

$$\mathbf{R}_s^b = \begin{bmatrix} \cos\beta_x \cos\beta_z & \cos\beta_y \sin\beta_z & -\sin\beta_y \\ \sin\beta_x \sin\beta_y \cos\beta_z - \cos\beta_x \sin\beta_z & \sin\beta_x \sin\beta_y \sin\beta_z + \cos\beta_x \cos\beta_z & \sin\beta_x \cos\beta_y \\ \cos\beta_x \sin\beta_y \cos\beta_z + \sin\beta_x \sin\beta_z & \cos\beta_x \sin\beta_y \sin\beta_z - \sin\beta_x \cos\beta_z & \cos\beta_x \cos\beta_y \end{bmatrix} \quad \text{eq. 4}$$

Additional information of the derivation and use of  $\mathbf{l}_b$  and  $\mathbf{R}_s^b$  in the geolocation and ortho-rectification process is provided in Section A-1 and A-2.

#### 4.2.3 Complete camera model

The complete camera model consists of eq. 1 through eq. 4. Let us represent this model as the function:

$$\mathbf{e}_{i,j}^b = M(x_i, y_j, C_{1,2,3,\dots,N}) \quad \text{eq. 5}$$

where  $C_{1,2,3,\dots,N}$  represents the camera model parameters listed in Table 5.

**Table 5 Camera model parameters**

| Symbol | Parameter                 | Representative Value |
|--------|---------------------------|----------------------|
| $f$    | focal length              | 7.02745 cm           |
| $x_0$  | x center point offset     | 0.00432022 cm        |
| $y_0$  | y center point offset     | -0.00897371 cm       |
| $x_L$  | x lever arm value         | 0.057 cm             |
| $y_L$  | y lever arm value         | 0.182 cm             |
| $z_L$  | z lever arm value         | 0.100 cm             |
| $W$    | camera to POS yaw angle   | -0.143 deg           |
| $\mu$  | camera to POS pitch angle | 0.385 deg            |
| $\xi$  | camera to POS roll angle  | 0.273 deg            |

#### 4.2.4 Calculating the camera parameters

Let the vector  $M$  represent the camera model parameters:

$$M = (f, x_0, y_0, x_L, y_L, z_L, \xi, \mu, W) \quad \text{eq. 6}$$

The camera model parameters are calculated by comparing the reported location of the same object in two separate images and adjusting the model parameters to minimize the difference. Two different types of comparisons are employed:

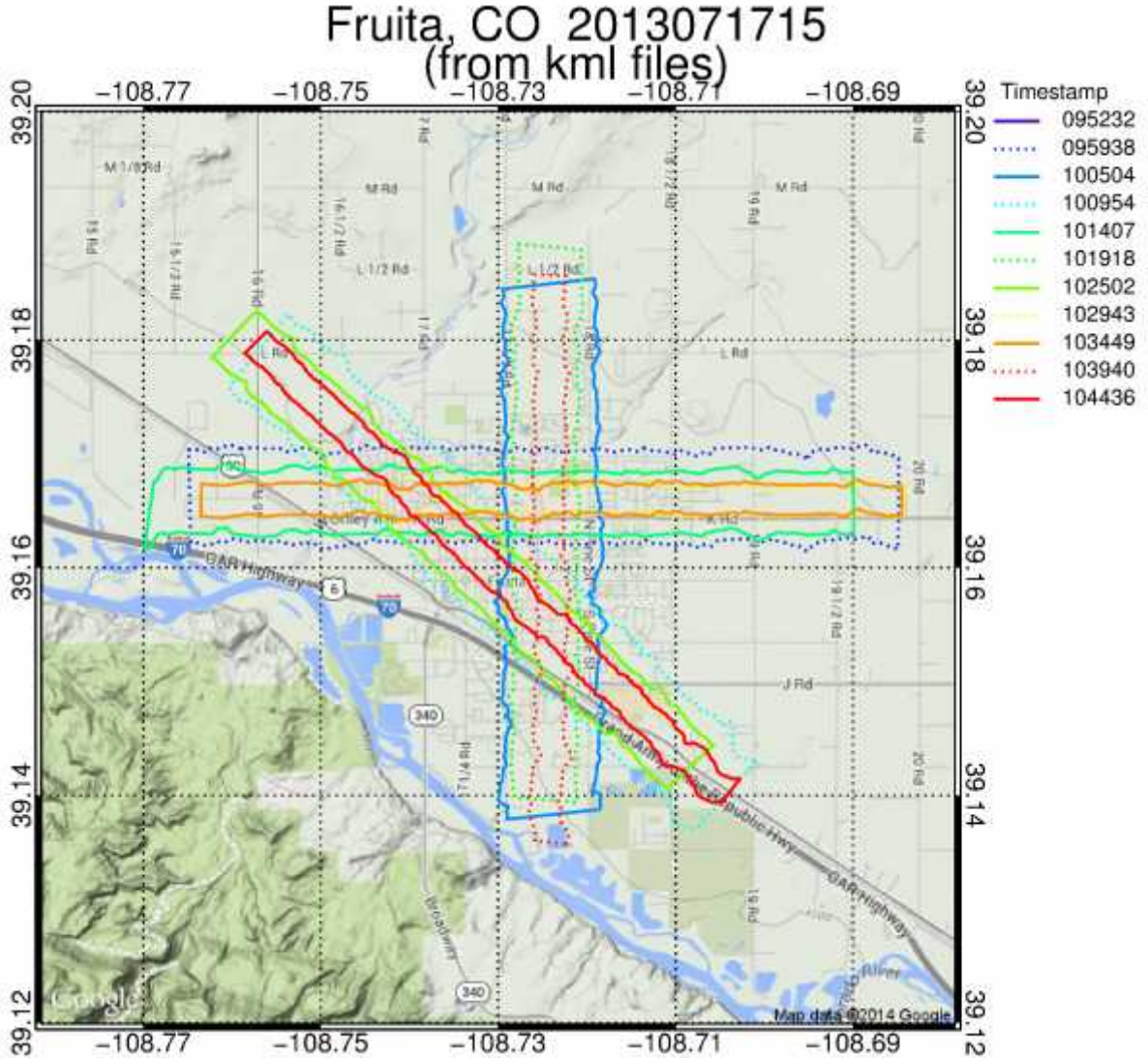
1. Ground control points (GCP's)
2. Tie points (tp's)

##### 4.2.4.1 Ground control points

GCP's are points in an image whose absolute location is known accurately in the  $m$ -frame, either by ground survey or by coincident LiDAR observations. They are created by matching an object in a camera image with the corresponding object in a geolocated LiDAR intensity image. This image, hereafter called the "LiDAR image", is the geolocated intensity of the LiDAR return and provides the 'ground truth' location data. The object (e.g., a street intersection, the corner of a roof, a manhole cover) in the camera image is matched with the corresponding object in the LiDAR image to the nearest pixel. This matchup provides the absolute location of the selected camera pixels.

To generate GCPs, first a set of camera images is orthorectified using a nominal camera model, e.g., last year's model. The images are chosen from a flight specifically designed for calibration. It includes overlapping flight tracks taken at different altitudes, from opposite directions, and crossing each other obliquely. From six to ten separate images are used to generate the necessary statistics. Figure 5 shows the outlines of the flight tracks for the calibration flight over Fruita, CO on 07/17/2013. The outlines

enclose the coverage of the spectrometer but are indicative of the camera coverage also. The width of the flight track is indicative of the flight altitude: the flights were at 500 m, 1000 m and 1500 m.

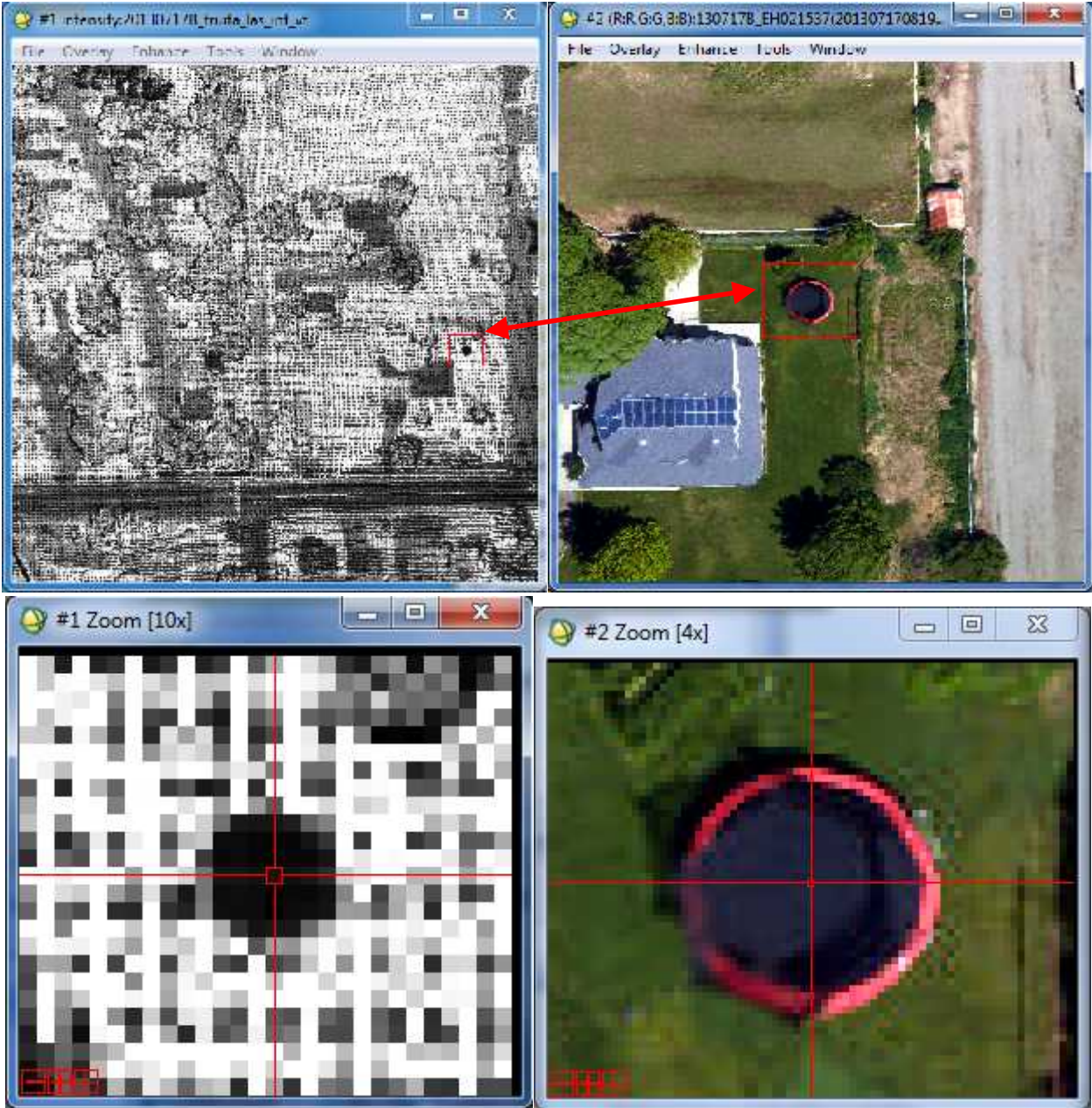


**Figure 5. Outline of the flight tracks over Fruita, CO, 07/17/2013**

The object matchup is performed by hand: an operator runs a program<sup>3</sup> that displays the LiDAR image and an overlapping camera image simultaneously. The operator selects a pixel in each image that represents the same object. The program records the pixel indices in the camera image in the s-frame, along with the corresponding pixel indices and latitude and longitude (lat/lon) values in the LiDAR image. The process is repeated for a number of objects (~25) distributed across the camera image.

<sup>3</sup> See Section 5.2.2.1 for the details of the programs used here

Figure 6 shows an example of a LiDAR image and a camera image viewing the same object, in this case an outdoor trampoline. In the top row, an arrow connects the corresponding objects while in the bottom row the cross hairs are centered on the object center. The output of the program is a file containing the matched pixel column and row numbers in the LiDAR and the camera images for each selected object and the lat/lon values for the LiDAR pixels.



**Figure 6. LiDAR intensity image and overlapping camera image showing a matching object: top row left-LiDAR image, top row right-camera image, bottom row, corresponding zoomed images centered on an outdoor trampoline**

Note however, that the camera image used in the matchup has been orthorectified using a nominal camera model. The pixel locations in the orthorectified files are then traced back to the pixel locations in the raw images. The result is a GCP file containing a table of the raw camera image sample and line indices  $i, j$  and the corresponding absolute location. Figure 7 shows the results for the images in Figure 6.



|                              |                                    |                         |
|------------------------------|------------------------------------|-------------------------|
| <b>LiDAR Intensity Image</b> | <b>Orthorectified Camera Image</b> | <b>Raw Camera Image</b> |
| Latitude = 39.164837 N       | $i = 555$                          | $i = 1515$              |
| Longitude = 108.703402 W     | $j = 2364$                         | $j = 1665$              |

**Figure 7. LiDAR image, orthorectified camera image and raw camera image centered on the same object. The text below the images gives the coordinates of the cross-hairs: lat/lon for the LiDAR image, sample (i) and line (j) indices of the camera images**

This process is repeated for a number of different camera images resulting in a separate GCP file for each camera image.

#### 4.2.4.2 Tie points

Tie points are the pixels in two different overlapping camera images that represent the same object, but for which accurate lat/lon information may not be known. An automated procedure is used to identify multiple tie points in overlapping images. For each tie point, the location of the object on the ground is calculated separately in each image using the nominal camera model. The model parameters are then adjusted to minimize the RMS error in the locations. Figure 8 shows two overlapping images and a tie point.



**Figure 8. Two overlapping digital images showing a tie point: full images, b. zoom level 1, c. zoom level 2. The red boxes enclose the same areas in each image. The red arrows connect the tie point, a white object on top of a roof.**

The automated procedure to generate tie points is implemented in the program “OpenSurf” (ED[06]). It can generate hundreds of tie points from a pair of overlapping images. However, it can also generate false matches: identifying two different objects as the same. Most of these false matches can be rejected by comparing the locations in the two images calculated by the nominal camera model.

Locations that differ by more than some threshold are discarded. Currently this threshold is 4 pixels = 32 cm.

#### 4.2.4.3 Parameter fitting

Given the GCP and tie points, we generate an optimal set of camera model parameters by minimizing the least squares error of the calculated locations of the combined GCP and tie points with respect to the model parameters. For the GCP points, the minimization is with respect to the calculated location versus the known location from the LiDAR image. For the tie points, the minimization is with respect to the locations calculated separately in the two images. Let:

$r_i = (x_i, y_i)$  = horizontal location vector (x and y only) in the m-frame

$\Delta r_i^g$  = location error for the  $i^{\text{th}}$  GCP point = difference between the calculate and the measured location

$\Delta r_j^t$  = location error for the  $j^{\text{th}}$  tie point = the difference between the calculated location in the two different images

$$R^g = \sqrt{\sum_{i=1}^{N_g} (\Delta r_i^g)^2 / N_g} = \text{root-mean-square GCP location error}$$

$$R^t = \sqrt{\sum_{j=1}^{N_t} (\Delta r_j^t)^2 / N_t} = \text{root-mean-square GCP location error}$$

$$RMS^{all} = (w^g RMS^g + w^t RMS^t) / (w^g + w^t) = \text{total weighted RMS location error}$$

where  $w^g, w^t$  are the relative weights given to the GCP and tie points respectively and  $N_g$  and  $N_t$  are the total number of GCP points and tie point respectively.

The location errors and hence  $RMS^{all}$  are functions of the camera model parameters. We use the Marquardt-Levenberg non-linear optimization algorithm (Press, 2007) to minimize the  $RMS^{all}$  with respect to the model parameters  $M$ . Typically, we use equal numbers of tie and control points (e.g., 25 for each image used) and  $w^g$  and  $w^t$  are both set to one. A single calibration with 10 overlapping flight images might then be seeking to minimize error of 250 ground control points along with 250 tie points. We typically achieve an RMS error for tie points in the 25 cm range with somewhat higher RMS error values for the ground control points.

Figure 9 shows the residual position errors for a sample camera model. In this case, the top plot shows a scatter plot of the error in x versus the error in y for both the GCP (red crosses) and the tie points (blue x's). The bottom plot shows the histograms of the positional errors. The mean of residual position errors

are about 10 cm for the tie points and 50 cm for the GCP points. The reason the GCP errors are larger is that the GCP points are matches between locations in the camera image, at 4 cm resolution, and in the LiDAR image, at 50 cm resolution. Therefore, the GCP points have a built-in uncertainty of  $\pm 25$  cm before adding the error in selecting points. Note that the x and y errors appear uniformly distributed around 0: there are no apparent systematic trends.

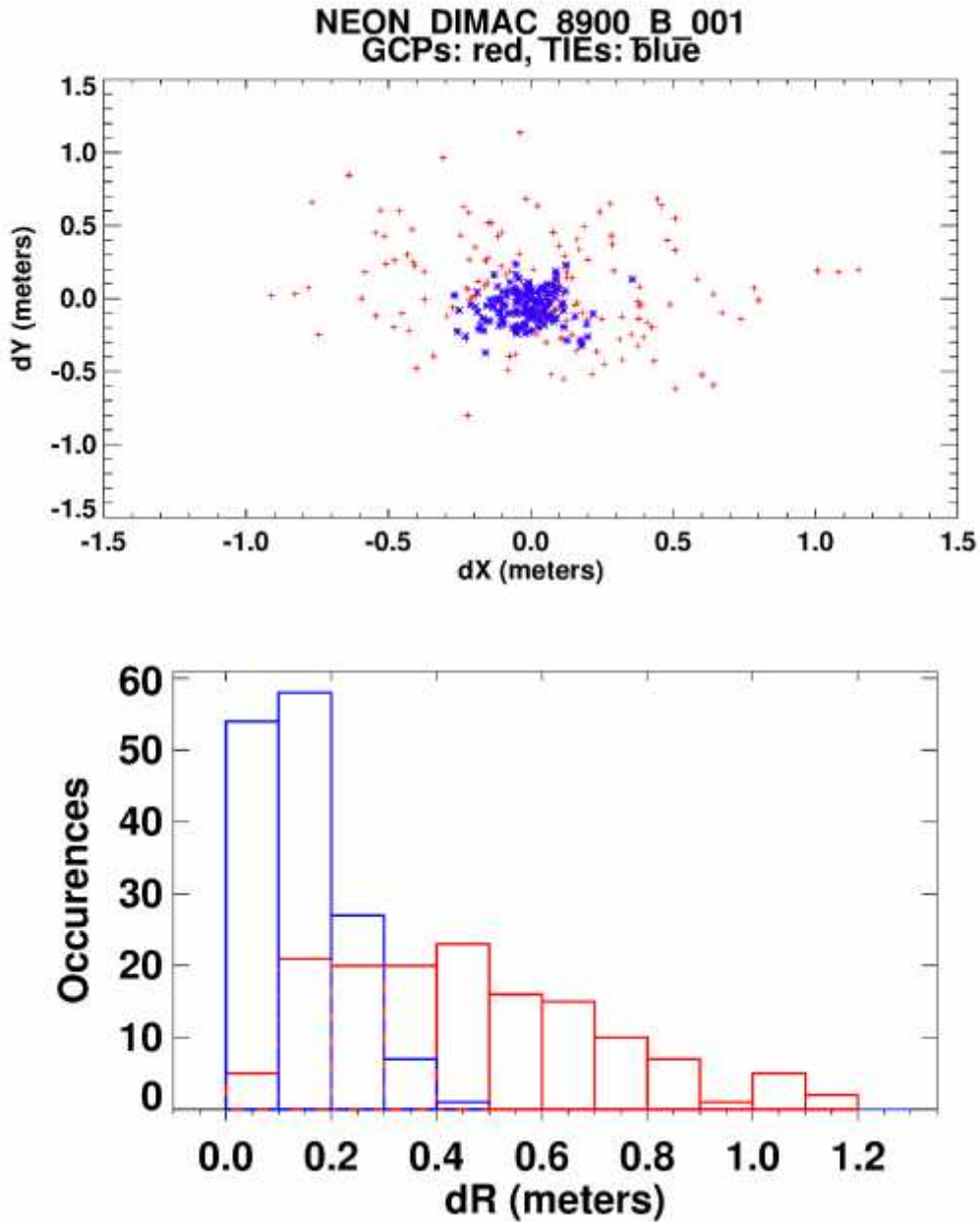


Figure 9. Location error for camera model NEON\_DiMAC\_8900\_B\_001. Top: scatter plot of error in x vs error in y, Bottom: histograms of the radian error. GCP points are in red, tie points are in blue.



|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

The process of creating the camera model must be repeated for each digital camera installation on an AOP payload. Note that the camera model should only change when the alignment between the LiDAR and the camera changes, e.g., when the camera and/or the LiDAR is physically removed and re-installed on the AOP Platform Integration Mount (PIM).

## 4.2.5 Orthorectifying the Corrected Images

### 4.2.5.1 Overview

In this step, the pixels from the original image projection are mapped onto a regular mapping (easting, northing) grid using the camera model generated in the previous step. The desired result is a set of images on a common grid that can be mosaicked to form a larger image with wall-to-wall coverage of the site.

Figure 10 shows an idealized sketch of the camera viewing the ground. The camera CCD sensor plane consists of a regular array of detectors. The line-of-sight (LOS) of each detector projects through the camera lens and intersects the ground. However, the detector intersections on the ground do not form a regular grid. This is because the variations in the terrain height and in the pointing of the camera cause the spacing of points on the ground to be irregular, and the orientation not aligned N-S (or E-W.)

The first step in creating the orthorectified image is to trace the edge pixels of the sensor plane down to the ground in a process termed ‘shooting-down’ (Appendix A-1). The next step is to construct the smallest regular grid that completely encloses the image outline, as shown in Figure 11. The grid is then filled in by tracing a ray from the center of each image cell back through the focal point up to the camera sensor CCD in a process termed ‘shooting up’ (Appendix A-2). The image cell is filled using nearest neighbor of the values on the detector surrounding the intersection point. Image cells outside the camera pixel outline shown in Figure 11 are filled with a value representing missing data.

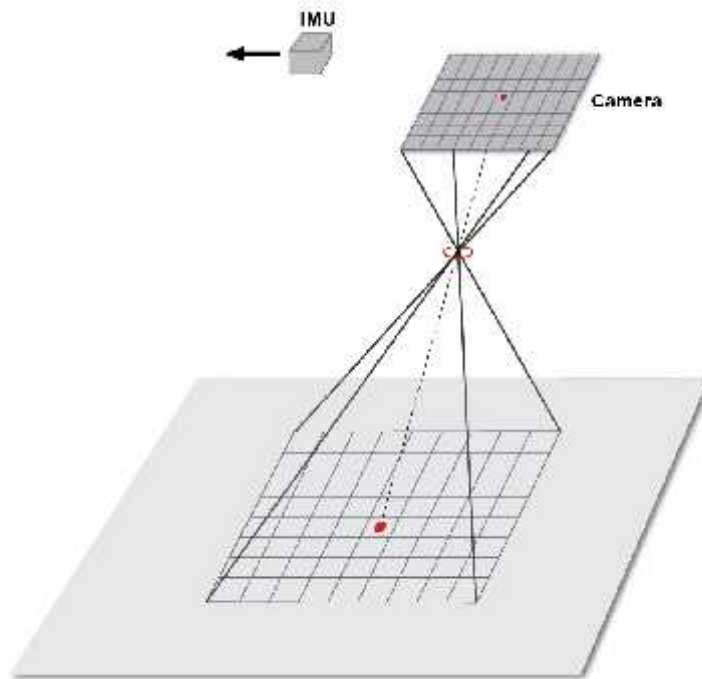


Figure 10. Sketch of camera above the ground showing the detector grid and the image grid on the ground

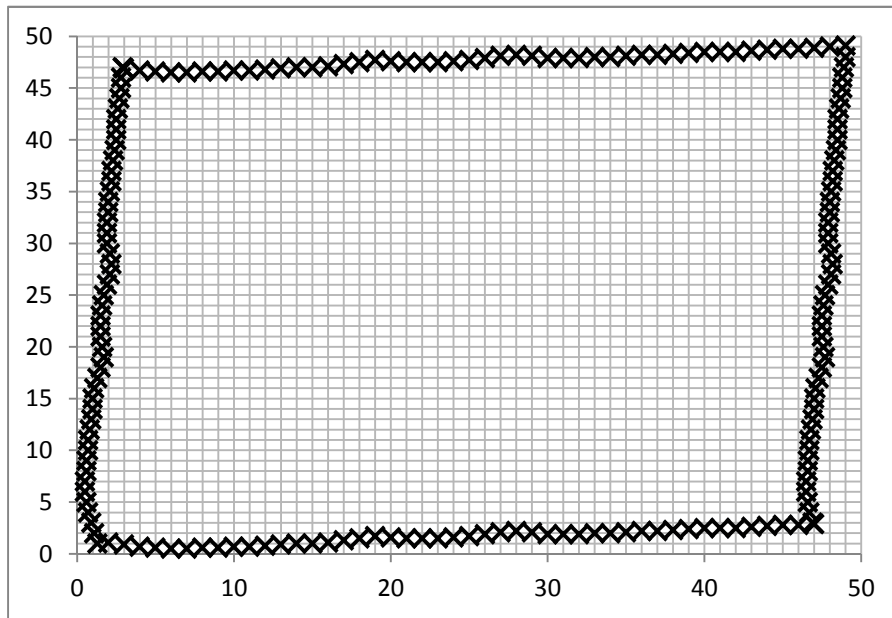


Figure 11. Sketch of a 50 by 50 image grid on the ground showing the locations where the camera edge pixels (X) intersect the ground.

|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |

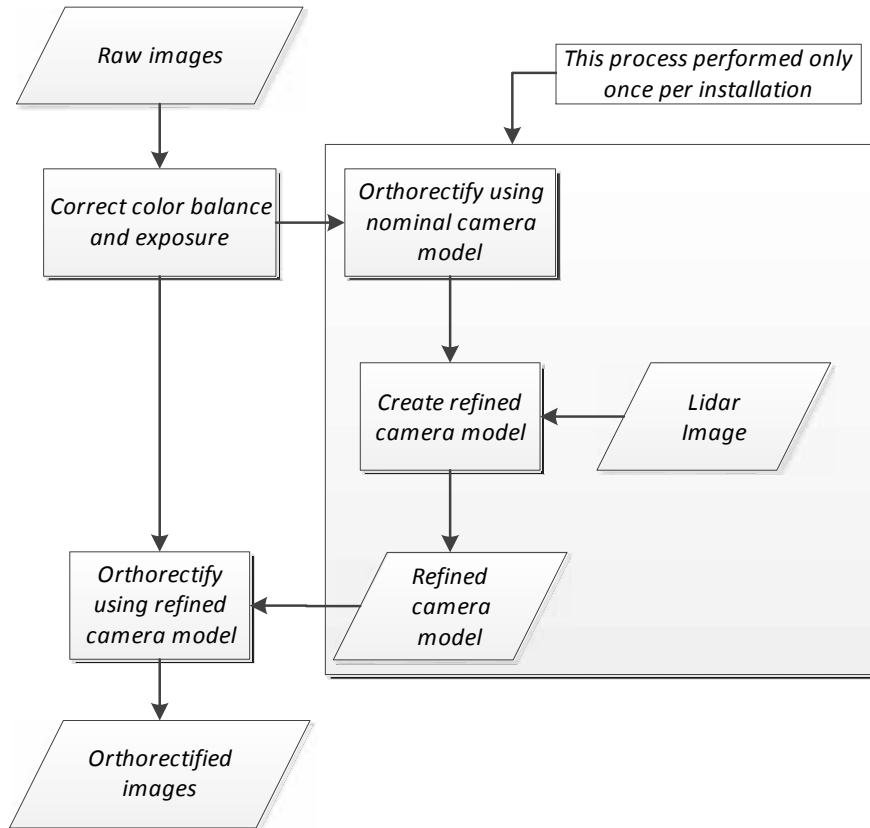
The mathematics of tracing the camera pixel line of sight (LOS) is described in Section A.1 and A.2.

**4.2.5.2 Spatial resolution: special considerations**

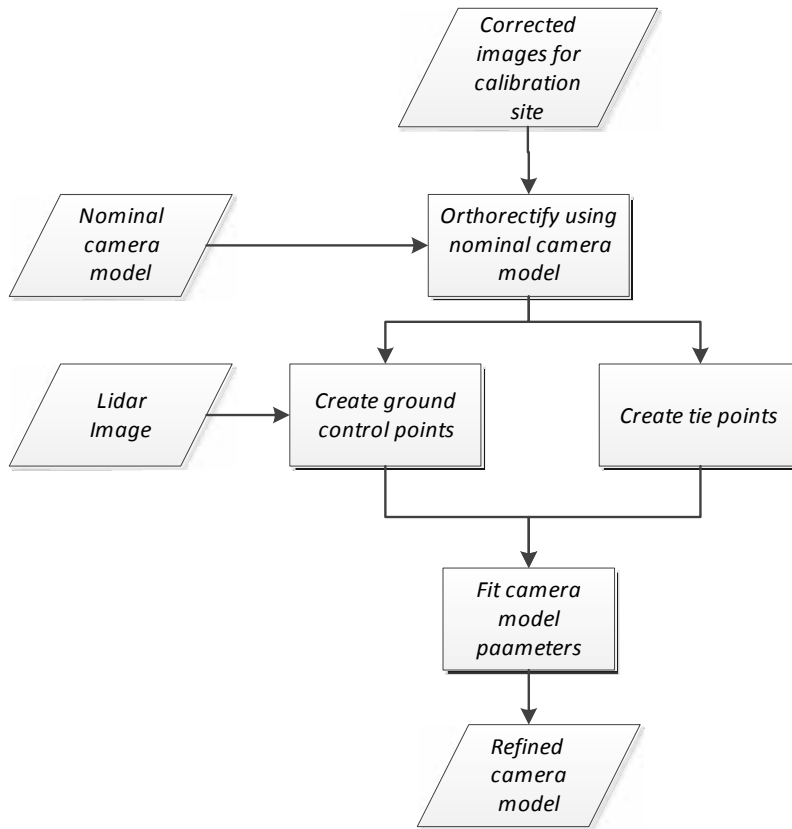
When flown at the nominal AOP elevations (e.g., 1000 m and 1500 m AGL), the digital camera produces imagery with nominal pixel center spacings of nominally 8 and 12 cm. However, the LiDAR data at these same altitudes is an order of magnitude coarser in spatial resolution. As such, one is faced with a difficult choice on how to handle this mismatch. Ray tracing the full-resolution image onto the LiDAR first return digital surface model (DSM) seems an obvious first choice but can lead to disconcerting imagery anomalies in areas with large vertical displacements such as tree branches and leaves, power lines, fence rails and flag poles. Alternatively, one can ray trace imagery using a “flattened” LiDAR elevation terrain model (DTM), where the trees and high curvature objects are removed. This has its own repercussions, as there will be a radial parallax pattern of “leaning out” of the real objects that are not represented in the smoothed elevation surface. A compromise involves spatially resampling digital camera image to reduce the LiDAR-to-camera resolution discrepancy. This is the approach that is being applied to the NEON orthorectified digital camera product. If we consider an image acquired at the nominal 1000 m flight altitude, each pixel ground sampling distance is 8 cm square. During processing, pixels are aggregated by a factor of 3.125 to a 25 cm aggregated pixel (at this flight altitude).

## 5 ALGORITHM IMPLEMENTATION

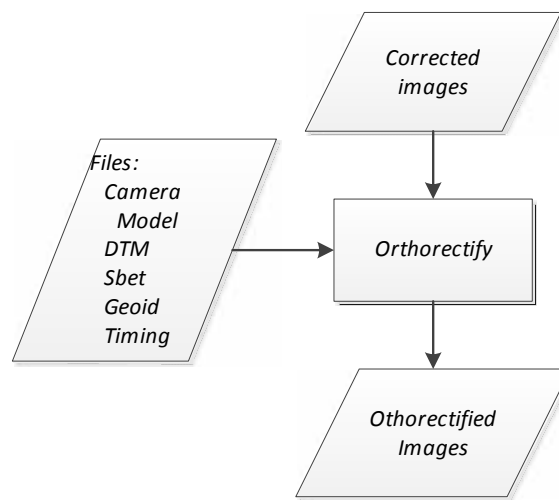
The overall digital camera orthorectification flowchart is shown in Figure 12: this chart shows the complete process of converting raw digital camera images to orthorectified images. Recall that the process of creating the refined camera model (outlined in a box) is performed only once per installation. The process of creating the refined camera model is expanded further in Figure 13 and the orthorectification process itself is shown in Figure 14.



**Figure 12. Digital camera orthorectification top-level flow chart**



**Figure 13. Camera model creation flow chart**



**Figure 14. Camera image orthorectification flow chart**

The processes described here are performed using commercial software “CameraOne” (ED[02]) and a set of IDL programs and procedures. We will describe the workflow required to perform each process but not the internal workings of the IDL programs. The IDL programs referenced here can be obtained from the Subversion (SVN)<sup>®</sup>(ED[07]) repository at NEON:

[https://svn.neon.local/svn/AOP/DataProcessing/OrthoProcessing/NIS\\_geolocate\\_calibrate/trunk](https://svn.neon.local/svn/AOP/DataProcessing/OrthoProcessing/NIS_geolocate_calibrate/trunk).

The programs can be found in the subdirectory “software/programs”.

The directory structure for the level 0 and 1 data files is described in AD[05], the NEON Data Products Naming Convention.

## 5.1 Color balance and exposure correction


### 5.1.1 Requirements



**Table 6. Color balance and exposure correction: required inputs and parameters**

| Input             | Reference | Description  |
|-------------------|-----------|--|
| <b>Files</b>      |           |  |
| raw camera images |           | raw camera images, in TIFF format. The images contain a 561 by 420 pixel thumbnail |
| <b>Parameters</b> |           |  |
| NA                |           | White balance and exposure are currently adjusted by an operator                   |

### 5.1.2 Processing steps

Currently color balance and exposure correction are performed using the software “[Capture One 7](#)” by [Phase One](#). The following is an outline of the workflow.

- 1) Open the program “Capture One”
- 2) From the “File” menu, select “New Catalog”. Supply a catalog name and a default folder location
- 3) From the “File” menu, select “Import Images”. Navigate to the directory containing the raw images. Thumbnails of the imported images will appear in the thumbnail window. Click on a thumbnail to display the corresponding image in the window above.
- 4) From the thumbnails of the imported images, select an image containing an white object, e.g., a white car or roof top.
- 5) From the toolbar, select the “Pick White Balance (W)” tool .
- 6) Hover the tool pointer over the white object. The red, green, blue and grey values of the pixel under the pointer are displayed on the bar just above the image. These values are between 0 and 255. Select a pixel with high values (>250) and Left-Click. The white balance adjustment is now saved.
- 7) From the “Edit” menu, select “Select All”. This selects all the images in the thumbnail window.

- 8) From the “Adjustments” menu, select “Apply Adjustments”. This applies the white balance adjustment to all the selected images.
- 9) From the thumbnails, select an image to use for the exposure correction. This image should contain a high dynamic range (both dark and light areas) to ensure that the exposure correction applied to other images does not render them over or under exposed.
- 10) From the toolbar, select the exposure tool 
- 11) From the “Exposure” tool bar, select the “Auto Adjust Exposure” tool 
- 12) From the “Edit” menu, select “Select All”. This selects all the images in the thumbnail window.
- 13) From the “Adjustments” menu, select “Apply Adjustments”. This applies the exposure correction to all the selected images
- 14) From the “File” menu, select “Process”. All of the selected images will be written to the output directory with the white balance and the exposure correction applied. These files will be in uncompressed tif format and will be about 173 MB (181,724,162 bytes) in size.

## 5.2 Camera model creation

### 5.2.1 Requirements

**Table 7. Camera model creation: required inputs and parameters**

| Input   | Reference   |
|---|-------------|
| <b>Files</b>  |             |
| LiDAR intensity image   | RD[04]      |
| Raw camera images after white balance and exposure correction | Section 5.1 |
| <b>Parameters</b>   |             |
| Nominal camera model  | Table B-2   |
| Sbet file   | RD[04]      |
| DEM file  | RD[04]      |
| Geoid file  | ED[01]      |

### 5.2.2 Processing steps

#### 5.2.2.1 Generate GCP points and tie points

- 1) Select calibration images
  - a) Select a flight specifically designed for creating both NIS and digital camera models.
  - b) Select from 6 to 10 images pairs
    - i) Pairs must overlap
    - ii) Include images from all altitudes
    - iii) Include images taken from opposite directions

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

- iv) Include images taken from crossing directions (e.g. N-S and E-W) if possible
- c) Orthorectify the camera images using a nominal camera model
  - i) See Section 5.3.2 for this procedure.
  - ii) Use a nominal camera model (e.g., last year's) to obtain an approximate orthorectification.
  - iii) Set the final pixel size to 4 cm for all images regardless of the flight altitude above ground.
- 2) Generate tie points
  - a) Run the program "aig\_neon\_dc\_build\_tp.pro" to generate tie points.
    - i) Run the IDL program `run_aig_neon_dc_build_tp.pro` to generate a set of tie points between overlapping images.
    - ii) See the comments in the header of this file for the details.
- 3) Generate GCP points
  - a) Run the ENVI program "Map->Registration->Select GCPs: image to image" to generate the ground control points.
    - i) This is a manual operated, operator intensive program
    - ii) For each selected image
      - (1) Match points between the LiDAR intensity image and a camera image
      - (2) Select about 25 points from each image
      - (3) Distribute the points around the camera image.
      - (4) Save the GCP's to a file.
      - (5) The process may take several hours for each image.
- 4) Run the IDL program "run\_aig\_neon\_dms\_summary.pro"
  - a) This program creates the image timing file which gives the gps seconds-of-week time and location for each of the images for this flight
  - b) See the comments at the beginning of the program for details

### 5.2.2.2 Calculate the camera model parameters

- 1) Run the IDL program "run\_neon\_aig\_dms\_cal\_GCP\_and\_tp" to generate a new camera model
  - a) Generate an input control file from the template file shown in Table 8
  - b) The user will be prompted to select the input control file, the GCP files and the tie point files.
- 2) Examine the residuals and the model parameters to validate the results
  - a) Compare the output residual plot with Figure 9. Residuals should be of the same magnitude
  - b) Check for systematic residuals. Trends may indicate that distortion parameters should be included in the camera model.
  - c) Compare the output model parameters with those in Table B-2, particularly the basic camera parameters. Large differences may indicate an error.

**Table 8. Template input control file for the program run\_neon\_aig\_dms\_cal\_GCP\_and\_tp.pro**

```
;;Template file for the inputs to the program run_neon_aig_dms_cal_GCP_and_tp.pro
; modelName : string = the name of the camera model to be created
```



```

; demfFile : string = the name of the dem file, fully qualified
; geoidFile : string = the name of the geoid file, fully qualified
; UTM_zone : integer = the UTM zone
; sumf : string = the name of the camera image timing and location file

modelName = 'NEON_DIMAC_8900_B_003.txt'
demFile = 'E:\AOP_data\D8900_camera_model\20130717B_fruita_las_dem_vf_bf'
geoidFile = 'E:\AOP_data\egm\GEOID09_conus_AIGLLC'
UTM_zone = 12
sumf = '\GCP_tie_pts\fruita_20130717b_summary.txt'

```

### 5.3 Camera image orthorectification

#### 5.3.1 Requirements

The required inputs for this algorithm are listed in Table 9. There are other optional parameters described but not currently used.

**Table 9. Camera image orthorectification: required inputs and parameters**

| Input             | Reference   | Description   |
|-------------------|-------------|---|
| <b>Files</b>      |             |   |
| camera image      | Section 5.1 | camera image, after white balance and exposure adjustment, in uncompressed tif format with an extension of '.tif' |
| sbet              | RD[04]      | trajectory of the platform: x, y, z, roll, pitch, yaw vs. time  |
| DSM               | RD[04]      | digital surface map   |
| Camera log file   |             | Log file containing image name and corresponding gps time   |
| camera model      |             | created in Section 5.2  |
| geoid             | ED[01]      | geoid file  |
| <b>Parameters</b> |             |   |
| pixel size        | 25 cm       | pixel size of the output image (m)  |

#### 5.3.2 Processing steps

The following list describes the orthorectification procedure.

- 1) Create a filtered DiMacLog file
  - a) This file contains a list of all the raw tif files and their corresponding gps seconds-of-week (sow) time. The data is extracted from the DiMac log file found in the level 0 data directory with a name similar to:  
[Mission\(ITCUBE\)\\_130717B\\_2013\\_07\\_17@06\\_52\\_21.tx](#)
  - b) Run the program [DiMACLogEx.exe](#)  
[\(https://svn.neon.local/svn/AOP/DataProcessing/OrthoProcessing/di](https://svn.neon.local/svn/AOP/DataProcessing/OrthoProcessing/di)

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

`sk_extract`) with this file as an argument to get the file:

`DiMACLogEx_Mission(ITCUBE)_130717B_2013_07_17@06_52_21.txt`

- 2) Create an input control file
  - a) Generate an input control file from the template file shown in Table 10. The file lists both required and optional parameters with the default values of the optional parameters shown.
- 3) Run the program “`run_neon_aig_vswir_geolocate.pro`”
  - a) The user will be prompted for the input control file

**Table 10 Template input control file for the program `run_neon_aig_dms_ortho.pro`**

```

;Template for the input file for the program run_neon_aig_dms_ortho.pro

; *****Required parameters*****
; flight
flight = '2013071715'

; dimac log file
dimaclog_file =
'Z:\flight\2013071715_AOP1\L0\Camera\DATALOG\MISSIONS\DiMACLogEx_Mission
(ITCUBE)_130717B_2013_07_17@06_52_21.txt'

; sbet file
sbet_file = 'Z:\flight\2013071715_AOP1\L1\trajectory\sbet_2013071715_V01.out'

; dem file
dem_file = 'D:\nleisso\2013071715_AOP1\20131010\20130717B_fruita_las_dem_vf_bf'

; utm zone
zone_in = 12

; geoid09 aigllc geoid file
geoid_file = 'z:\external\egm\GEOID09_conus_AIGLLC'

; camera model
camera_model = 'NEON_DIMAC_8900_B_001'

; names of the tif files to process:
; -enter '' to be prompted for the files,
; -enter 'all' to process all the tif files in the data directory
tiff_files = 'all'

; *****Optional parameters***** (values shown are the defaults)
; data directory: location of the tif files
data_dir = '..\output\'

; output directory
output_dir = '..\'

; dem null value
dem_null_in = -9999.

; ps flag (-9999. uses data-derived value per image)
ps_in = 0.25d

; glt flag (0 do not create, 1 do create)
glt_flag = 1

; edge flag (0 do not create, 1 do create)

```



|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |

edge\_flag = 1

|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |

## 6 UNCERTAINTY

### 6.1 Analysis of uncertainty

Since the digital camera is the highest resolution sensor on-board the AOP, it will collect imagery whose precision outstrips the accuracy of the required auxiliary data sets providing position, attitude, timing and surface elevation. As such, it is difficult to define precisely the ultimate accuracy of the orthorectification of the digital imagery since it is a combined effect of LiDAR resolution, as discussed below, and the camera data themselves. In the future, we plan to establish accurately located reference points observable by both the LiDAR and the digital camera. These points will allow us to estimate the accuracy of the geolocation with respect to ‘ground truth.’

### 6.2 Reported uncertainty

See Section 4.2.4.3.

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## 7 VALIDATION AND VERIFICATION

It is important to appreciate that the absolute error in the camera and spectrometer data from “truth” (in this case, the measured GPS points) has two components: 1) the LiDAR geolocation error, and 2) the camera (or spectrometer) co-registration to the LiDAR data. The first term cannot be corrected in the camera (or spectrometer) geolocation processing, since the LiDAR image is used as truth. Therefore, the accuracy of the camera and spectrometer geolocation processing can only be measured against the LiDAR data.

### 7.1 Data product validation

Validation of image geolocation accuracy will be determined by comparison to ground control points as measured with ground-based GPS ground stations.

### 7.2 Data product verification

We plan to establish several ground points that can be observed in the imagery and obtain accurate GPS measurements of these points. Flights could then be made at several different altitudes and directions to capture this ground reference points in the imagery. The data would then be processed and the geolocation error determined in a) LiDAR data, b) spectrometer data, and c) camera data.

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## 8 FUTURE MODIFICATIONS AND PLANS

### 8.1 Establish accurately measured ground control points

We plan to establish several ground points that can be observed in the imagery and obtain accurate GPS measurements of these points. Flights could then be made at several different altitudes and directions to capture this ground reference points in the imagery. The data would then be processed and the geolocation error determined in a) LiDAR data, b) spectrometer data, and c) camera data.

### 8.2 Move from proprietary to non-proprietary software

Currently, the white balance and exposure corrections are performed using the program “[Capture One 7](#)” (ED[02]) by [Phase One](#) (ED[04]). In the future, we plan to evaluate non-proprietary software for these functions (e.g. [Raw Therapee](#) (ED[05])) and develop an automated procedure that does not require a man-in-the-loop for processing.

### 8.3 Refine the resampling procedure

Currently, the nominally 8 cm resolution images are resampled to 25 cm by selecting the last pixel to fall into the 25 cm grid box. This procedure discards about 85% of the original data. Consideration will be given to modify the resampling algorithm with a more robust rebinning algorithm that preserves more of the original data.

Resampling to the 25 cm ground sampling resolution has been implemented to eliminate image anomalies as described in Section 4.2.5.2. Any rebinning algorithm considered for implementation in the future would need to adequately address this issue.

### 8.4 Mosaic and tile the images

Software is under development to mosaic and tile the camera images. The resulting images will be much easier to navigate and to co-locate with the spectrometer images.



|  |                           |                            |
|--|---------------------------|----------------------------|
| <i>Title:</i> NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                           | <i>Date:</i> March 2, 2015 |
| <i>NEON Doc. #:</i> NEON.DOC.001211  | <i>Author:</i> W. Gallery | <i>Revision:</i> A         |

## 9 CHANGE LOG

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

## 10 REFERENCES

- Brown, Dwane C. 1965. "Decentering Distortion and the Definitive Calibration of Metric Cameras." In *Annual Convention of the American Society of Photogrammetry, Washington, D. C., March 1965*. Washington, DC. [https://eserv.asprs.org/PERS/1966journal/may/1966\\_may\\_444-462.pdf](https://eserv.asprs.org/PERS/1966journal/may/1966_may_444-462.pdf).
- . 1971. "Close-Range Camera Calibration." In *Symposium on Close-Range Photogrammetry*. Urbana, IL. [www.vision.caltech.edu/bouguetj/calib\\_doc/papers/Brown71.pdf](http://www.vision.caltech.edu/bouguetj/calib_doc/papers/Brown71.pdf).
- "Levenberg–Marquardt Algorithm - Wikipedia, the Free Encyclopedia." 2014. Accessed June 23. [http://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt\\_algorithm](http://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt_algorithm).
- Press, William H., ed. 2007. *Numerical Recipes: The Art of Scientific Computing*. 3rd ed. Cambridge, UK ; New York: Cambridge University Press.
- Schenk, T. 2014. "Introduction to Photogrammetry." Accessed June 17. <http://www.mat.uc.pt/~gil/downloads/IntroPhoto.pdf>.



## APPENDIX A MATHEMATICS OF GEOLOCATION

The geo-referencing and ortho-rectification of camera images (Figure 1) are primarily performed through translations and rotations of the LOS of the camera CCD sensor elements through a series of Cartesian coordinate systems. This appendix describes the coordinate systems used in the geolocation and the methods used to transform the location of an object from one coordinate system to another to provide a geo-referencing solution. It is intended to provide an overview of the geolocation process with sufficient detail to enable the reader to create code to implement the process from raw data. As mentioned in Section 4.2.2, the optics of the camera system is simplified to a ‘pin-hole’ camera model for the purposes of the theoretical treatment of geo-referencing and ortho-rectification (Figure 4). The first section describes geolocation by directing the LOS toward the ground and intersecting with a DTM, also termed ‘shooting-down’. The second section describes tracing a ray from the ground location to a specific sensor element, termed ‘shooting up.’ As described in section 4.2.5, both methods are used in the geolocation and ortho-rectification of the camera imagery. Shooting-down is implemented to determine where the perimeter pixels of the image intersect the ground (Figure 11), while shooting-up determines where the interior pixels intersect the CCD frame.

### A.1 Geolocation for the shooting-down method

In order to achieve a geo-referenced pixel from an acquired digital image in a real-world coordinate system using the shooting-down method, we must first describe a vector which defines the location of any arbitrary pixel in the image. This is accomplished by first defining the pixel frame ( $g$ ), a two-dimensional frame, with origin at the upper left corner of the image, positive  $x$  direction toward the right of the image, and positive  $y$  toward the bottom of the image. Our definition of the  $g$ -frame is selected to follow the traditional photo coordinate system of standard aerial film cameras (Abdullah et al., 2013). Within the  $g$ -frame, any given pixel can be described with a vector  $\mathbf{u}_g$  as

$$\mathbf{u}_g = \begin{bmatrix} x_g \\ y_g \\ 0 \end{bmatrix} \quad (1)$$

where  $x_g$  and  $y_g$  represent the row and column offsets from the origin of the  $g$ -frame. Next, we determine the offset from  $\mathbf{u}_g$  to the images principal point location ( $pp$ ) and apply a correction for radial lens distortion (if it exists), as follows

$$\mathbf{u}_{p-g} = \begin{bmatrix} x_g - x_p - dx_r \\ y_g - y_p - dy_r \\ 0 \end{bmatrix} \quad (2)$$

where  $dx_r$  and  $dy_r$  are the  $x$  and  $y$  components of the radial distortion respectively. Vector  $\mathbf{u}_{p-g}$  is transferred from the two-dimensional  $g$ -frame to the image coordinate frame ( $i$ ), a three dimensional Cartesian frame. The  $i$ -frame has origin at the perspective center of the camera lens, a positive  $x$ -axis

parallel with the flight direction,  $y$ -axis toward the left side of the image, and  $z$ -axis pointing up. The transformation to the  $i$ -frame is accomplished through a reorientation of the axis of the  $g$ -frame with a matrix  $\mathbf{R}_{p-g}^i$ , and inserting the focal length ( $f$ ) to the  $z$ -dimension of  $\mathbf{u}_{p-g}$  as follows

$$\mathbf{u}_i = \mathbf{R}_{p-g}^i \begin{bmatrix} x_g - x_p & -dx_r \\ y_g - y_p & -dy_r \\ & -f \end{bmatrix} \tag{3}$$

where

$$\mathbf{R}_{p-g}^i = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

Digital Camera sensor frame

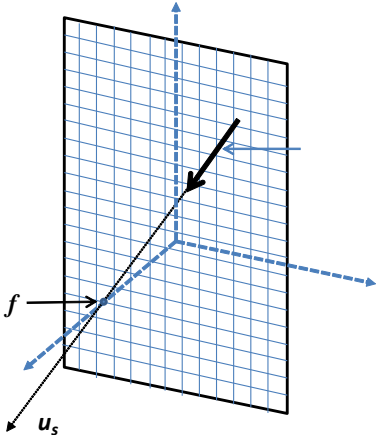


Figure A-1 Camera CCD sensor related to the  $i$ -frame

Once in the  $i$ -frame,  $\mathbf{u}_i$  is then transferred into the sensor frame ( $s$ ), which represents how the imaging sensor is mounted in the airborne platform. For the digital camera, the  $s$ -frame is defined by a two dimensional CCD array. The  $s$ -frame has origin at the perspective center of the camera lens, positive  $x$ -direction parallel to the direction of flight, positive  $y$ -axis to the right of the CCD sensor and positive  $z$ -axis direction downward (Figure A-1). To transfer  $\mathbf{u}_i$  to the  $s$ -frame the axes must be re-oriented with a matrix  $\mathbf{R}_i^s$  as follows

$$\mathbf{u}_s = \mathbf{R}_i^s \mathbf{u}_i \tag{5}$$

where

$$\mathbf{R}_i^s = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \quad (6)$$

Once in the sensor frame,  $\mathbf{u}_s$  can then be transformed into the body frame ( $b$ ). The  $b$ -frame definition is specific to the manufacture and installation of the inertial measurement unit (IMU). Applanix, the providers of the IMU used in the AOP payload, define the axes of their IMU systems to have an x-axis with positive direction toward the nose of the aircraft, y-axis with positive direction toward the right wing of the aircraft and z-axis pointing downward (Applanix, 2012). The downward direction forms the primary reference for the right-handed system, as it will be coincident with the direction of local gravity when the plane is perfectly level. The relationship between the  $s$ -frame and the  $b$ -frame is theoretically a static one, described by three Euler angles commonly referred to as boresight angles ( $\beta$ ). The values of  $\beta$  are determined from an in-flight calibration procedure, which is performed a minimum of once per payload installation. Knowledge of  $\beta$  allows  $\mathbf{u}_s$  to be transferred to the  $b$ -frame as follows

$$\mathbf{u}_b = \mathbf{R}_s^b \mathbf{u}_s \quad (7)$$

where

$$\mathbf{R}_s^b = \mathbf{R}_{\beta_x} \mathbf{R}_{\beta_y} \mathbf{R}_{\beta_z} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta_x & \sin\beta_x \\ 0 & -\sin\beta_x & \cos\beta_x \end{bmatrix} \begin{bmatrix} \cos\beta_y & 0 & -\sin\beta_y \\ 0 & 1 & 0 \\ \sin\beta_y & 0 & \cos\beta_y \end{bmatrix} \begin{bmatrix} \cos\beta_z & \sin\beta_z & 0 \\ -\sin\beta_z & \cos\beta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (8)$$

and  $\beta_x$ ,  $\beta_y$ , and  $\beta_z$  are rotations about the x-axis, y-axis and z-axis of the  $s$ -frame respectively. Eq. 8 can be combined and written as

$$\mathbf{R}_s^b = \begin{bmatrix} \cos\beta_x \cos\beta_z & \cos\beta_y \sin\beta_z & -\sin\beta_y \\ \sin\beta_x \sin\beta_y \cos\beta_z - \cos\beta_x \sin\beta_z & \sin\beta_x \sin\beta_y \sin\beta_z + \cos\beta_x \cos\beta_z & \sin\beta_x \cos\beta_y \\ \cos\beta_x \sin\beta_y \cos\beta_z + \sin\beta_x \sin\beta_z & \cos\beta_x \sin\beta_y \sin\beta_z - \sin\beta_x \cos\beta_z & \cos\beta_x \cos\beta_y \end{bmatrix}. \quad (9)$$

The orientation of the IMU frame is an important consideration during the transfer of  $\mathbf{u}_s$  to the  $b$ -frame as ensuring the appropriate sign on the rotations is critical. Since the  $s$ -frame is a right-handed system, boresight angles with positive sign will initiate clockwise rotations when looking down the positive direction of each respective axis. Therefore, when determining the boresight angles from the in-flight calibration procedure, care must be taken to ensure the rotations are produced with the appropriate sign. It should also be noted that boresight angles are typically small, often only fractions of a degree. In this situation we can approximate  $\cos(\beta) \approx 1$  and  $\sin(\beta) \approx \beta$ , which allows a simplification of Eq. 9 to:

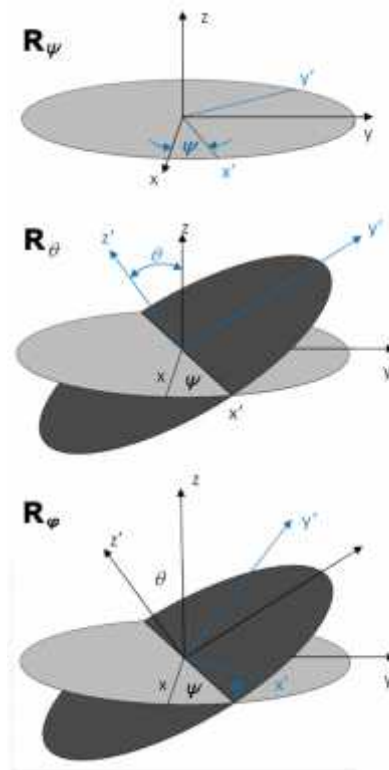
$$\mathbf{R}_s^b = \begin{bmatrix} 1 & \beta_z & -\beta_y \\ -\beta_z & 1 & \beta_x \\ \beta_y & \beta_x & 1 \end{bmatrix}$$

Once in the body frame,  $\mathbf{u}_b$  can be transferred to the navigation frame ( $n$ ). The  $n$ -frame has positive x-direction toward geodetic north (true north), y-axis with positive direction toward east and z-axis with positive direction toward the local ellipsoidal normal (Figure A-3). This frame is also commonly referred to as a NED (north-east-down) geographic frame (see Abdullah et al., 2013). The transfer of  $\mathbf{u}_b$  to the  $n$ -frame requires the roll ( $\varphi$ ), pitch ( $\theta$ ) and yaw ( $\psi$ ) values as observed by the IMU. Similar to the boresight angles, the sign of these measurements as recorded by the IMU is an important consideration. According to the definition of the  $b$ -frame described by Applanix (2012), positive angular changes will result in roll when the aircraft's right wing tips down, in pitch when the aircraft's nose tips up, and in heading when the aircraft makes a right-hand turn. To transfer  $\mathbf{u}_b$  from the  $n$ -frame to the  $b$ -frame, a similar relationship as shown in (8) and (9) can be used, with values for roll, pitch and yaw substituted for the boresight angles. This combination follows the traditional aerospace sequence (Kuipers, 1999) of Euler angle rotations (Figure A-2), and can be written as:

$$\mathbf{R}_n^b = \mathbf{R}_\varphi \mathbf{R}_\theta \mathbf{R}_\psi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

which can be combined to:

$$\mathbf{R}_n^b = \begin{bmatrix} \cos\varphi\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \sin\varphi\cos\theta \\ \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi & \cos\varphi\cos\theta \end{bmatrix}. \quad (11)$$



**Figure A-2 – The three rotations used in the aerospace sequence to transfer a vector from the *b*-frame to the *n*-frame**

However, note that this provides a rotation from the *n*-frame to the *b*-frame, where we desire the rotation from the *b*-frame to the *n*-frame, i.e. the inverse rotation is required. Since rotation matrices of Euler angles are orthogonal, the combined rotation matrix is also orthogonal and for any orthogonal matrix it holds true that:

$$\mathbf{R}^T = \mathbf{R}^{-1} \tag{12}$$

from which it follows that (Jekili, 2000, pg. 11):

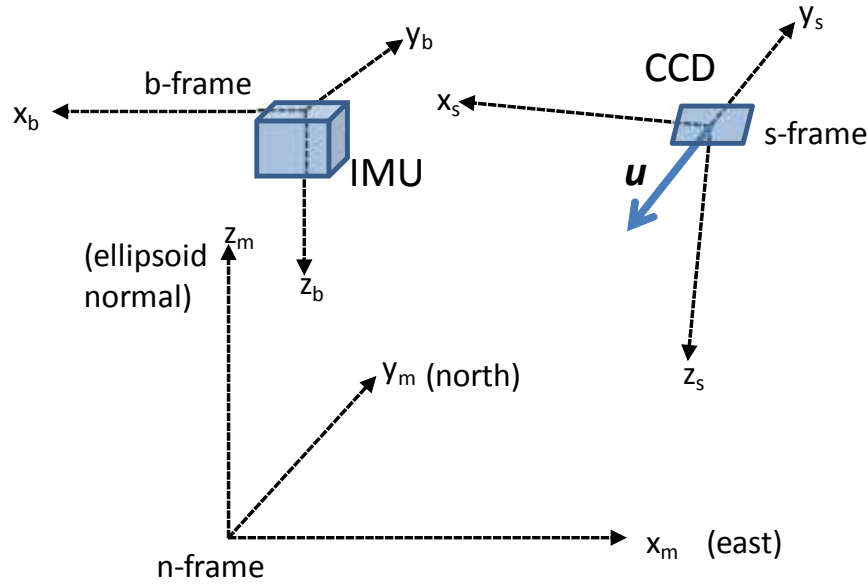
$$(\mathbf{R}_\psi \mathbf{R}_\theta \mathbf{R}_\phi)^{-1} = (\mathbf{R}_\psi \mathbf{R}_\theta \mathbf{R}_\phi)^T. \tag{13}$$

This relationship can be exploited to achieve the inverse rotation of  $\mathbf{R}_{n}^b$  by applying the transpose operation; therefore:

$$\mathbf{R}_b^n = [\mathbf{R}_n^b]^T \begin{bmatrix} \cos\varphi\cos\psi & \sin\varphi\sin\theta\cos\psi - \cos\varphi\sin\psi & \cos\varphi\sin\theta\cos\psi + \sin\varphi\sin\psi \\ \cos\theta\sin\psi & \sin\varphi\sin\theta\sin\psi + \cos\varphi\cos\psi & \cos\varphi\sin\theta\sin\psi - \sin\varphi\cos\psi \\ -\sin\theta & \sin\varphi\cos\theta & \cos\varphi\cos\theta \end{bmatrix}. \quad (14)$$

and it follows that:

$$\mathbf{u}_n = \mathbf{R}_b^n \mathbf{u}_b. \quad (15)$$



**Figure A-3 Relationship between the s-frame, b-frame and n-frame**

Ultimately, we desire  $\mathbf{u}$  to be in a global reference frame, as opposed to the local  $n$ -frame. To convert to a global mapping frame, we first determine  $\mathbf{u}_n$  in an ECEF (Earth centered earth fixed) frame. The ECEF frame is a Cartesian frame with origin at the center of a reference ellipsoid, positive direction of the  $x$ -axis toward the intersection of the equator and Greenwich meridian,  $y$ -axis with positive direction toward the intersection of the equator and 90° east meridian, and  $z$ -axis with positive direction through the geographic northpole. The transfer of  $\mathbf{u}_n$  to ECEF coordinates is performed as follows:

$$\mathbf{u}_E = \mathbf{R}_n^E \mathbf{u}_n \quad (16)$$

where, following Jekeli (2000, pg. 25),

$$\mathbf{R}_n^E = \begin{bmatrix} \cos(-\lambda) & 0 & -\sin(-\lambda) \\ 0 & 1 & 0 \\ \sin(-\lambda) & 0 & \cos(-\lambda) \end{bmatrix} \begin{bmatrix} \cos(\pi/2 + \phi) & \sin(\pi/2 + \phi) & 0 \\ -\sin(\pi/2 + \phi) & \cos(\pi/2 + \phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$$= \begin{bmatrix} -\sin\phi\cos\lambda & -\sin\lambda & -\cos\phi\cos\lambda \\ -\sin\phi\sin\lambda & \cos\lambda & -\cos\phi\sin\lambda \\ \cos\phi & 0 & -\sin\phi \end{bmatrix}$$

where  $\phi$  and  $\lambda$  are the latitude and longitude of the principal point sensors perspective center. At this stage,  $\mathbf{u}_G$  is has origin at the camera CCD sensor and goes through the focal point, however, the absolute location of the aircraft given by the SBET is at a reference point on the IMU. The coordinates must be transferred from the IMU reference point to the camera focal point ( $O$ ). Let the location of  $O$  be described with  $\mathbf{v}$ . The translation distances between these two locations is physically measured in the lab and are termed the ‘lever arms’ (see Section 4.2.2.3). The translation of coordinates from the IMU reference location to  $O$  is done as follows:

$$\mathbf{v}_E = \mathbf{R}_n^E \left( \mathbf{l}_b + \mathbf{R}_n^b \mathbf{R}_E^n \mathbf{s}_E \right) \quad (18)$$

where  $\mathbf{s}$  is the location of aircraft at the IMU reference point, and  $\mathbf{l}$  is the measured lever arm offsets (Figure A-4). Following this,  $\mathbf{v}_E$  can be transferred to  $\mathbf{v}_G$  with equations 18-21. The ECEF coordinates can be translated into latitude and longitude to put  $\mathbf{v}_E$  into a geographic frame ( $G$ ) using well known relationships (Jekeli, 2001, pg.24), as follows:

$$\mathbf{u}_G = \begin{bmatrix} \phi \\ \lambda \\ h \end{bmatrix}_G = \begin{bmatrix} \tan^{-1} \left( \frac{z_E}{\sqrt{x_E^2 + y_E^2}} \left( 1 + \frac{e^2 N \sin \phi}{z_E} \right) \right) \\ \tan^{-1} \left( \frac{y_E}{x_E} \right) \\ \frac{\sqrt{x_E^2 + y_E^2}}{\cos \phi} - N \end{bmatrix} \quad (19)$$

where  $h$  is the ellipsoidal height measured along the ellipsoidal normal,  $N$  is the radius of curvature of the reference ellipsoid in the prime vertical plane, determined as

$$N = \frac{a}{\sqrt{1 - e^2 \sin^2 \phi}} \quad (20)$$

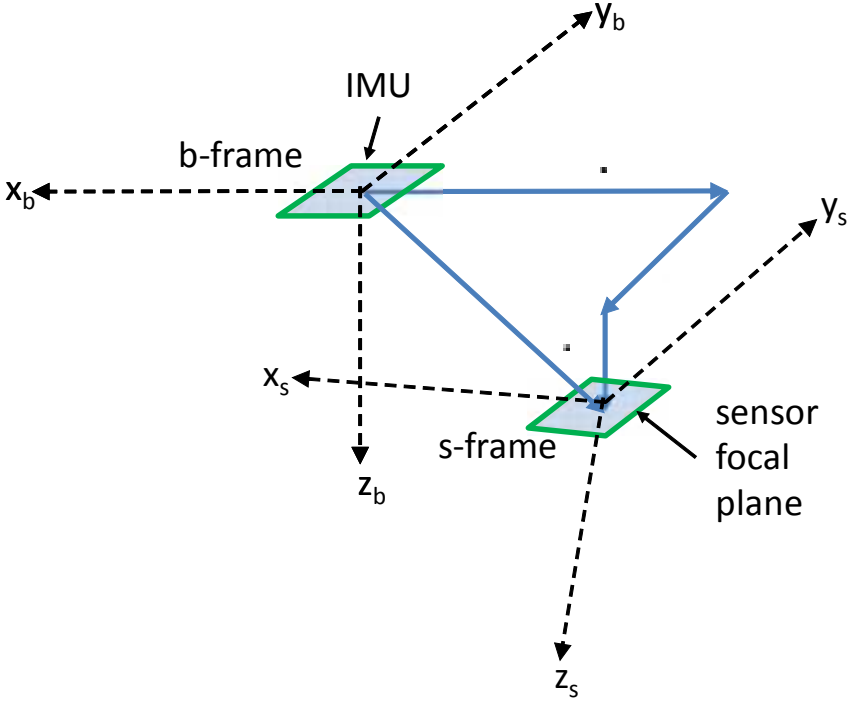
where  $a$  is the semi-major axis of the reference ellipsoid, and  $e$  is the eccentricity which is determined as

$$e^2 = 2f - f^2 \quad (21)$$

and  $f$  is the flattening for the specific desired reference ellipsoid. For reference, the values for the semi-major axis and flattening of the GRS80 ellipsoid, which is used in the WGS84 datum, are 6378137 m and 1/298.257222101 respectively. The solution for  $\phi$  in (18) is not a closed form solution, and an iterative procedure must be employed to achieve a result of sufficient accuracy. This requires an initial estimate of  $\phi$ , where  $h$  is assumed to be zero, determined as follows

$$\phi_U \equiv \phi(h = 0) = \tan^{-1} \left( \frac{z_E}{\sqrt{x_E^2 + y_E^2}} \left( \frac{1}{1 - e^2} \right) \right) \quad (22)$$

and through each successive iteration, (19) and (20) must be updated. This leaves  $\mathbf{v}_G$  and  $\mathbf{u}_G$  in geographic coordinates, which can be converted to UTM coordinates for ray-tracing  $\mathbf{u}$  to a DSM. To convert geographic coordinates to UTM coordinates well-known formulations from Snyder (1989) can be implemented, and these are not necessarily repeated here. The direction indicated by  $\mathbf{u}_E$  can be applied to  $\mathbf{v}_E$  to provide the direction of the LOS in the ECEF frame. If the vector is extended, the end point can be transformed to UTM coordinates with equations 19 through 22. Once the vector is established in the UTM mapping frame it can be ray-traced onto a DTM (also in the UTM mapping projector

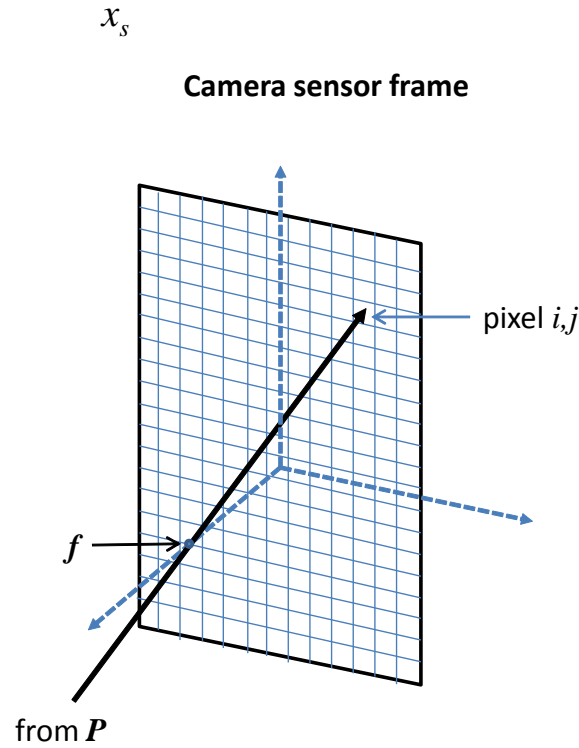


**Figure A-4 - Translational relationship between the *b*-frame and the *s*-frame through the three components of the lever arm vector between the IMU and the sensor focal plane.**



## A.2 Shooting up

To associate an observation made by a specific sensor on the CCD camera sensor array to a ground pixel location, a method referred to as ‘shooting-up’ is employed. The shooting-up process involves tracing the direction of the optical path of each sensor element back through several of the direct georeferencing equations described in Section A.1 and associating it with a specific sensor element (Figure A-5).



**Figure A-5 LOS of sight from ground location to sensor pixel location in the shooting-up method**

To initiate the process, each pixel in the camera image must have a known three-dimensional coordinate in the Universal Transverse Mercator mapping frame. The coordinate is required in the UTM frame because the LiDAR DSM used for determining the pixel elevations will have reference to the UTM frame in the typical NEON workflow. However, any other arbitrary mapping frame is acceptable if the elevation model shares the same coordinate reference. To determine the coordinate of each image pixel in the UTM frame, the shooting-down method (Section 4.2.4 and A-1) is employed to determine the coordinates of all pixels around the perimeter of the image (Figure 11). A grid is then filled inside of the perimeter of pixels, and an associated coordinate is defined based on the grid cell size. Once UTM coordinates are established in each grid cell, let the coordinated UTM location of an arbitrary pixel be described with a vector  $\mathbf{u}_m$ , where  $m$  designates the UTM mapping frame. The first step is to transform  $\mathbf{u}_m$  to the G-frame ( $\phi, \lambda$ , height). This is accomplished through well-known transformations, which have

been described in Snyder (1982). Following this, the coordinates are transformed to the ECEF-frame with a transformation as follows:

$$\mathbf{u}_E = \begin{bmatrix} (N(\phi) + h)c_i \\ (N(\phi) + h)c_i \\ (N(\phi)(1 - e^2) + h)s_i \end{bmatrix}. \quad (23)$$

Once  $\mathbf{u}_E$  is obtained, the direction in the ECEF frame between the pixel coordinate and the camera's principal point ( $O$ ) is required. A similar process is applied to achieve the three-dimensional location of  $O$  (designated with  $\mathbf{v}$ ) in the ECEF frame. To accomplish this, the lever arm offsets ( $\mathbf{l}$ ) between the SBET and  $O$  (measured in the lab) must be applied in the  $b$ -frame. Application of  $\mathbf{l}$  in the  $b$ -frame and conversion to the ECEF frame can be done using equations 14 through 17. The full transformation which determines  $\mathbf{v}_E$  can be written as follows:

$$\mathbf{v}_E = \mathbf{R}_n^E \mathbf{R}_b^n \mathbf{l}_b + \mathbf{s}_E \quad (24)$$

The direction of the vector between the pixel on the ground and  $O$  can be found through a differencing and normalization of  $\mathbf{u}_E$  and  $\mathbf{v}_E$  :

$$\mathbf{e}_E = \frac{\mathbf{v}_E - \mathbf{u}_E}{|\mathbf{v}_E - \mathbf{u}_E|} \quad (25)$$

where  $\mathbf{e}$  is the unit vector showing direction between the pixel coordinate and  $O$ . Following this,  $\mathbf{e}_E$  is rotated into the  $n$ -frame frame. This is accomplished through the rotation matrix  $\mathbf{R}_E^n$ , which can be calculated as  $[\mathbf{R}_n^E]^{-1}$ , where  $\mathbf{R}_n^E$  is given as equation 17:

$$\mathbf{e}_n = \mathbf{R}_E^n \mathbf{u}_E \quad (26)$$

Next, we must transfer  $\mathbf{e}_n$  into the body frame, using  $\mathbf{R}_n^b$  as defined in equation 11, as follows:

$$\mathbf{e}_b = \mathbf{R}_n^b \mathbf{u}_n \quad (27)$$

Following this,  $\mathbf{e}_b$  is rotated into the sensor frame by applying the boresight misalignments:

$$\mathbf{e}_s = \mathbf{R}_b^s \mathbf{e}_b \quad (28)$$

The rotation matrix  $\mathbf{R}_b^s$  reverses the rotation of  $\mathbf{R}_s^b$  shown in equation 8, 9 and 10. To determine  $\mathbf{R}_b^s$ , the boresight misalignments can be entered as negative values into 8, 9 or 10, or determined as  $[\mathbf{R}_s^b]^{-1}$ .

Once in the  $s$ -frame,  $\mathbf{e}_s$  can then be rotated into the  $i$ -frame as follows:

$$\mathbf{e}_i = \mathbf{R}_s^i \mathbf{e}_s \quad (29)$$

where  $\mathbf{R}_s^i$  is identical to  $\mathbf{R}_i^s$  given in equation 6. To use  $\mathbf{e}_i$  to determine the desired sensor element in the CCD array, the angle ( $\alpha$ ) between the unit vector and the z-axis of the  $i$ -frame is determined as:

$$\alpha = \cos^{-1} \left( \frac{\mathbf{e}_i \cdot \mathbf{f}_i}{|\mathbf{e}_i| |\mathbf{f}_i|} \right) \quad (30)$$

where:

$$\mathbf{f}_i = \begin{bmatrix} 0 \\ 0 \\ f \end{bmatrix} \quad (31)$$

The distance between  $O$  and the CCD sensor ( $x_{O \rightarrow C}$ ) along  $\mathbf{e}_i$  is then determined as:

$$x_{O \rightarrow C} = \frac{f}{\cos \alpha} \quad (32)$$

And it follows that the location of the sensor element on the CCD sensor ( $\mathbf{w}_i$ ) is:

$$\mathbf{w}_i = f x_{O \rightarrow C} \quad (33)$$

We are only interested in the first two elements of  $\mathbf{w}_i$  ( $x, y$ ), as the third element ( $z$ ) is equal to  $f$ . These coordinates are then determined in the  $g$ -frame by rotating  $\mathbf{w}_i$  by  $\mathbf{R}_i^g$  and applying the principal point offsets and distortion parameters (determined in the camera calibration) as follows:

$$\mathbf{w}_g = \mathbf{R}_p^i \begin{bmatrix} x_g - x_p & -dx_r \\ y_g - y_p & -dy_r \\ 0 & \end{bmatrix} \quad (34)$$

where  $\mathbf{R}_p^i$  is identical to  $\mathbf{R}_p^i$  given in equation 4. The  $x$  and  $y$  components of  $\mathbf{w}_g$  will then provide the appropriate CCD sensor location in the  $g$ -frame.

### A.3 Shooting down vs shooting up: tradeoffs

The choice of which approach to use is governed by the following tradeoffs:

|   |                    |                     |
|---|--------------------|---------------------|
| Title: NEON Algorithm Theoretical Basis Document (ATBD) - AOP Digital Camera Image Orthorectification |                    | Date: March 2, 2015 |
| NEON Doc. #: NEON.DOC.001211  | Author: W. Gallery | Revision: A         |

1. Shoot up is computationally much more efficient than shoot down: there is an analytical solution and no iterative algorithm is needed.
2. Shoot down better preserves the fidelity to the original data: each record (spectrum or image pixel) is individually honored and proper located.

The digital camera geolocation algorithm uses the shoot up approach while the spectrometer geolocation algorithm uses the shoot up approach.

#### A.4 Bibliography

Jekeli, C. (2001), *Inertial navigation systems with geodetic applications*, Walter de Gruyter, Berlin ; New York.

Kuipers, J. B. (1999), *Quaternions and rotation sequences: a primer with applications to orbits, aerospace, and virtual reality*, Princeton University Press, Princeton, N.J.

Snyder, J. P. (1987), *Map projections—A working manual*, 1395, USGPO.

## APPENDIX B BROWN'S CAMERA DISTORTION MODEL

The distortions associated with the digital camera are modeled using Brown's camera model (Brown 1971),(Brown 1965) are modeled using the following equation:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} x_u \\ y_u \end{bmatrix} \left( 1 + K_1 r^2 + K_2 r^4 + K_3 r^6 \right) + \begin{bmatrix} \left( P_1 (r^2 + 2x_u^2) + 2P_2 x_u y_u \right) \\ \left( P_2 (r^2 + 2y_u^2) + 2P_1 x_u y_u \right) \end{bmatrix} \left( 1 + P_3 r^2 + P_4 r^4 \right) \quad \text{eq. B-1}$$

where:

$(x_u, y_u)$  = the undistorted image point as projected by an ideal pin-hole

$(x_d, y_d)$  = the distorted image point as projected on the image plane

$K_n$  = the n'th radial distortion coefficient

$P_n$  = the n'th tangential distortion coefficient

$$r = (x_u^2 + y_u^2)^{1/2}$$

Table B-1 defines the camera model parameters while Table B-2 lists the parameters for three different camera models. The first model in Table B-2 is a nominal model used to perform the initial orthorectification of the images chosen to create the final camera model. The next two are the final models before and after August 2013 respectively. Note that only the first three camera parameters (C, Xp, Yp) are used. Experience has shown that including more parameters does not measurably improve the results.

**Table B-1. List of camera model parameters**

| Variable                          | Definition  |
|-----------------------------------|---|
| <b>Basic Camera Parameters</b>    |   |
|                                   | Equivalent Focal Length (cm)                                |
| (X <sub>0</sub> ,Y <sub>0</sub> ) | Principal Point Offset (cm)                                 |
| <b>Installation Parameters</b>    |   |
| (S2B1,S2B2,S2B3)                  | Sensor to Body angles: roll, pitch, yaw (deg)               |
| (LA1,LA2,LA3)                     | Lever arm, camera focal point from IMU reference point (cm) |
| <b>Brown Distortion Model</b>     |   |
| (K1,K2,K3)                        | Symmetric Radial Distortion Coefficients                    |
| (P1,P2)                           | Decentering Distortion Coefficients                         |
| (B1,B2)                           | Differential scaling and non-orthogonality Coefficients     |
| C'                                | Calibrated (Balanced) Focal Length                          |
| (K'0,...,K'4)                     | Balanced Symmetric Radial Distortion Coefficients           |

**Table B-2. Camera model parameters for three models**

| Parameter                          | Model Name               |                   |                   |
|------------------------------------|--------------------------|-------------------|-------------------|
|                                    | DIMAC 8900 initial model | DiMAC_89000_A_006 | DiMAC_89000_B_001 |
| <b>Basic Camera Parameters</b>     |                          |                   |                   |
| C (cm)                             | 7.0                      | 7.0360426666375   | 7.0274485031893   |
| Xp (cm)                            | 0.0                      | 0.0133882028640   | 0.0043202290385   |
| Yp (dm)                            | 0.0                      | -0.0021263642400  | -0.0089737077618  |
| <b>Installation Parameters</b>     |                          |                   |                   |
| S2B1 (deg)                         | 0.0                      | 0.272897397745    | 0.018523056100    |
| S2b2 (deg)                         | 0.0                      | 0.385281803849    | 0.397369122613    |
| S2b3 (deg)                         | 0.0                      | -0.143365695317   | 0.120484892925    |
| LA1 (cm)                           | 0.057                    | 0.057             | 0.057             |
| LA2 (cm)                           | 0.182                    | 0.182             | 0.182             |
| LA3 (cm)                           | 0.100                    | 0.100             | 0.100             |
| <b>Brown Distortion Parameters</b> |                          |                   |                   |
| K1                                 | 0.0                      | 0.0               | 0.0               |
| K2                                 | 0.0                      | 0.0               | 0.0               |
| K3                                 | 0.0                      | 0.0               | 0.0               |
| P1                                 | 0.0                      | 0.0               | 0.0               |
| P2                                 | 0.0                      | 0.0               | 0.0               |
| B1                                 | 0.0                      | 0.0               | 0.0               |
| B2                                 | 0.0                      | 0.0               | 0.0               |
| C`                                 | 0.0                      | 0.0               | 0.0               |
| K'1                                | 0.0                      | 0.0               | 0.0               |
| K'2                                | 0.0                      | 0.0               | 0.0               |
| K'3                                | 0.0                      | 0.0               | 0.0               |
| K'4                                | 0.0                      | 0.0               | 0.0               |