

A METADATA BASED APPROACH FOR ANALYZING UAV DATASETS FOR PHOTOGRAMMETRIC APPLICATIONS

A. Dhanda¹, F. Remondino², M. Santana Quintero¹

¹ Dept. of Civil and Environmental Eng., Carleton University, Ottawa, Canada – Email: abhudhanda@cmail.carleton.ca, mariosantana@cunet.carleton.ca

² 3D Optical Metrology (3DOM) unit, Bruno Kessler Foundation (FBK), Trento, Italy – Email: remondino@fbk.eu, web: <http://3dom.fbk.eu>

Commission II, ICWG I/II

KEY WORDS: UAV, Metadata, Filtration, Photogrammetry

ABSTRACT:

This paper proposes a methodology for pre-processing and analysing Unmanned Aerial Vehicle (UAV) datasets before photogrammetric processing. In cases where images are gathered without a detailed flight plan and at regular acquisition intervals the datasets can be quite large and be time consuming to process. This paper proposes a method to calculate the image overlap and filter out images to reduce large block sizes and speed up photogrammetric processing. The python-based algorithm that implements this methodology leverages the metadata in each image to determine the end and side overlap of grid-based UAV flights. Utilizing user input, the algorithm filters out images that are unneeded for photogrammetric processing. The result is an algorithm that can speed up photogrammetric processing and provide valuable information to the user about the flight path.

1. INTRODUCTION

Within the last year, Unmanned Aerial Vehicles (UAVs) platforms are becoming cheaper, more affordable and stable. We are witnessing an increasing use of these platforms for a variety of different tasks. UAVs can be classified either by their category of use, by their propulsion system, or by their aerodynamic features - lighter-than-air, rotary wing, and fixed wing (Nex and Remondino, 2014). UAVs can be used in a wide variety of fields: agriculture, forestry, archaeology, architecture, environmental monitoring, emergency response and assessment, and even traffic monitoring (Gomez and Perdue, 2016; Giordan et al., 2018). UAV flight patterns and image blocks differ according to planning and acquisition properties. They vary from very irregular image blocks to more constant and grid- or circular-base image blocks (Fig. 1). Grid-based UAV flights (Fig. 2) follow a consistent pattern, which makes it possible to analyse them quickly and consistently. Most grid-based UAV flights are flown with rotary wing UAVs, but they can also be flown with fixed wing UAVs although their higher instability can cause more irregularity in the block. Most common systems (i.e. platform, camera and base-control) create and store flight metadata, which is generally written into the image (EXIF and XMP) or in a separate file (XMP). Exchangeable Image File Format (EXIF) and Extensible Metadata Platform (XMP) are two of the more common formats found in images. Though they primarily contain different information it possible for an image to have EXIF and XMP tags that describe the same thing (Harvey, 2018). Metadata information has a *tag* and a *name*. When using a program like Exiftool to read the metadata of an image, it will present you with the metadata name and its value. However, to extract that data for processing needs, you need to know the tag. Metadata tags can be found by writing the metadata to a text file (Harvey, 2018). For example, 'Gimbal Yaw Degree' is a name, and 'XMP:GimbalYawDegree' is a tag.

1.1 Aim of the paper

The work aims to investigate the block geometry of typical grid-based UAV flights performed for 3D surveying and modeling

reasons. Such image blocks are normally acquired with a regular interval time (e.g. 1 sec) based on user specifications (e.g. along-track / end or across-track / side overlap, GSD, etc.), often producing an abundant number of images with respect to the necessary ones. This abundance is, from one side, increasing redundancy but, from the other side, it can affect the overall processing procedure, in terms of elaboration time and accuracy. We aim to exploit block metadata and tag information in order to achieve more regular image blocks. The proposed methodology filters out images to reduce large block sizes and speed up photogrammetric processing.

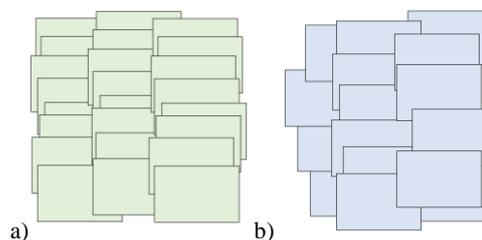


Figure 1. Example of UAV flight patterns: strip/grid-based image block (a) and irregular overlap (b).

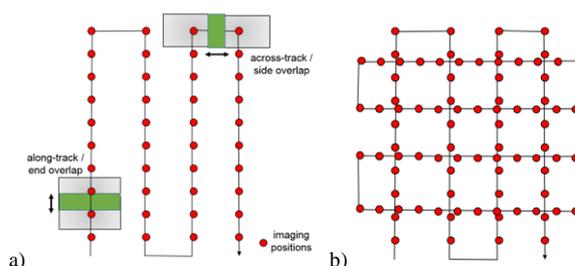


Figure 2. Along-track / end overlap and across-track / side overlap in an UAV flight path composed of parallel strips (a). A criss-cross flight plan (b).

2. RELATED WORKS

Extensive research has been done on UAV navigation and positioning based on Global Navigation Satellite Systems (GNSS) and Inertial Measurement Unit (IMU) observations (Colomina and Molina, 2014), visual systems (Smith et al., 1986; Leonard and Durrant Whyte, 1991) and a combination of the two (Wang et al., 2008). Bartelsen and Mayer (2010) use GPS values to upgrade relative to absolute orientation 3D models. GNSS/IMU-based navigation of low-cost UAVs is generally not accurate; the light weight and light payload result in sensor data that cannot reliably be used for photogrammetric georeferencing (Barazzetti et al., 2010; Lari et al., 2015). Direct georeferencing (DG) or direct sensor orientation of UAV images is still an open issue (Rehak et al., 2013; Turner et al., 2013; Eling et al., 2015; Tampubolon and Reinhardt, 2016; Benassi et al., 2017). According to Pfeifer et al., 2012, the GNSS and IMU measurements on certain low-cost UAVs can achieve georeferencing precision below 1 m, pitch and roll measurements with a precision of below 1 degree, and yaw measurements with a precision of 2.5 degrees. GNSS data is generally used for DG as initial approximation of the exterior orientation parameters.

3. METHODOLOGY

The proposed methodology takes the imagery and metadata tags, and analyses the overlap between images to detect over-redundancy and inconsistency in the block geometry. So far, a flat terrain is considered. The final output is a filtered number of images with more constant overlap to allow faster processing and better results. If the GNSS measurements are not directly embedded into the metadata, a simple script can be used to write such data into the image headers.

The methodology was developed in python 2.7; it is outlined in Figure 3 and covered in more detail in the following subsections (2.2 – 2.7). The code is available on Github.

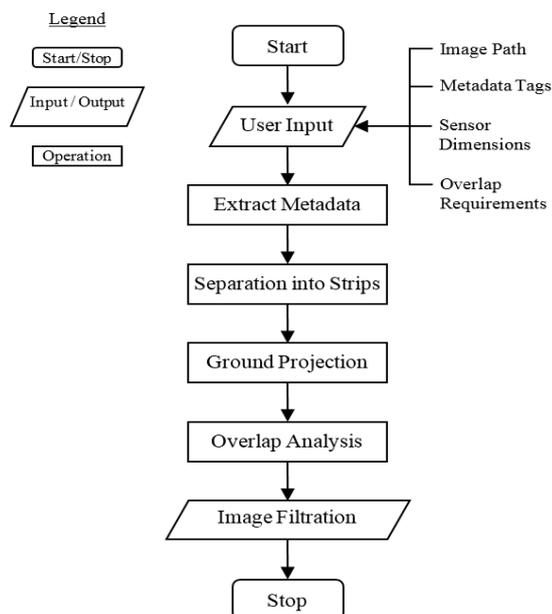


Figure 3. Methodology of the image filtering algorithm.

3.1 User Input

User input is needed for the information that is either not in the metadata or is required before the images are read: the path to the

image block, the length and width of the camera sensor, the relevant metadata tags, and the required end overlap and side overlap for the project (which could be different from the flown one). It should be noted that metadata are not always written consistently between different manufacturers. Therefore, the metadata tags for the required data need to be manually identified by the user. For example, DJI's Phantom 4 and Phantom 4 Pro write the metadata tags for the image width differently from each other: 'EXIF:ExifImageWidth' and 'EXIF:ImageWidth' respectively.

3.2 Extract Metadata

The metadata is obtained using a python wrapper for Exiftool, an open source software for extracting metadata from files (Harvey, 2018). The metadata tags for each parameter are stored into variables that are used throughout the algorithm. Table 1 shows the metadata tags used to pull the required information from an image taken by a DJI Phantom 4.

Tag	Tag Explanation
'EXIF:DateTimeOriginal'	The time that the image was taken
'EXIF:FocalLength'	The focal length of the UAV camera
'EXIF:GPSLatitude'	The latitude recorded by the onboard GPS
'EXIF:GPSLongitude'	The longitude recorded by the onboard GPS
'XMP:RelativeAltitude'	The altitude of the craft measured against the take off point
'XMP:FlightYawDegree'	The yaw angle of the UAV
'XMP:GimbalYawDegree'	The yaw angle of the camera gimbal
'XMP:GimbalPitchDegree'	The pitch angle of the camera gimbal
'XMP:GimbalRollDegree'	The roll angle of the camera gimbal

Table 1. Metadata tags from an image taken by a DJI Phantom 4 UAV

3.3 Separation into Strips

After the metadata is extracted from the block the images are sorted based on their capture time, and then each image is sorted into one of the flight's strips. Using the yaw (heading) of the craft, it is determined if an image is part of the same strip as the image before it. This is also used to differentiate strips that move along the same axes.

3.4 Ground Projection

The ground projection is calculated using the same principles as a projective transformation (Luhmann et al., 2011). The UAV with a forward facing camera position is used as the origin of a local coordinate system. First, the horizontal and vertical field of view (HFOV and VFOV) are calculated using sensor dimensions, and the camera focal length. Using the VFOV and HFOV, unit vectors are created in the direction of each of the four corners of the image, and then combined to create a polygon. A quaternion-based rotation matrix (Hamilton Quaternion) is created for each of the rotation angles (yaw, pitch, and roll) (Henderson, 1997; Horn, 1987). The rotation matrices are multiplied by the polygon to obtain the orientation of the camera.



Figure 4. Images from a DJI Phantom 4 from each of the test cases: Nora (a), Forte Mattarello (b), Moore Farm (c)

Then, a ground plane is created using the altitude of the UAV with respect to the ground. Rays are projected from the origin, through the corners of the image to the ground plane. It is from these intersection coordinates that the 'ground polygon' (or image footprint) is formed and used to calculate the overlap between images.

The reliability of this phase is limited by the precision of the onboard GNSS, in particular for low-cost UAVs (Lari et al., 2015).

3.5 Overlap Analysis

Using the GNSS based polygons created in 2.5 an intersection calculation is used to determine the overlapping area between different images. The sequential images in each strip are used to calculate the end overlap. If the end overlap between two images (image n and $n+1$) is greater than the required overlap, the first image is compared to another one two steps away from it (image n and $n+2$). If the overlap between those two is sufficient, then the image in between them ($n+1$) is marked as an image to be filtered. A strip-based approach is used for the side overlap calculations. For example, strip one is compared to strip two (the next strip with a heading along the same axis), and the greatest side overlap is determined for each image in strip one. If, as a whole, they exceed the required side overlap, then strip one is compared to strip three. If that side overlap meets the requirements, then strip two is marked as a strip to be filtered.

3.6 Image Filtration

The images marked to be filtered are moved to a subfolder created in the image directory. The results of the filtering procedure, and the final achieved overlap are written to a text file and displayed in a summary table.

4. CASE STUDY: OVERLAP OF LOW-COST UAVS

Test flights were flown with low-cost UAVS to help determine the reliability of the data gathered from inexpensive sensors. The flights were flown using a DJI Phantom 4 and the Pix4Dcapture app at three different sites in 2017. An image from each of the sites is shown in Figure 4: the archaeological site of Nora, Sardinia, Italy (a); a WWI fortification, Forte Mattarello, near Trento, Italy (b); the barn of the Moore Farm in Gatineau, QB, Canada (c). Flights from each location were run through the algorithm. The results (mean overlap and standard deviation from the mean) were calculated using python's numpy package. They were compared to the settings used in the app during the flight and are displayed in Tables 2-3-4.

The results from Nora and Forte di Mattarello are quite close to the expected values, except for the side overlap of Flight 3 from the Nora dataset. The Moore Farm results vary significantly from the expected values. The deviation could either be from the low-cost GNSS not following the flight pattern exactly, or a weak

signal causing the GNSS to incorrectly reflect the image overlap. A visual analysis of the images determined that it is most likely the latter. The former likely accounts for the large standard deviations in some of the flights. Figure 5 shows the inconsistent overlap of three images from Flight 1 of the Nora dataset. The red area of Figure 6a highlights the three images and shows that the overlap is represented accurately by the GNSS data.

	Nora					
Flight #	1		2		3	
# of Images	334		155		251	
Overlap	End	Side	End	Side	End	Side
Flight Settings (App) (%)	80	40	80	40	80	40
Algorithm Mean (%)	79	44	80	53	75	32
Algorithm Standard Deviation (%)	2.3	3.2	4.2	5.5	10.5	2.7

Table 2. Results from the Nora overlap tests.

	Forte di Mattarello			
Flight #	1		2	
# of Images	131		50	
Type of Overlap	End	Side	End	Side
Flight Settings (App) (%)	80	40	80	40
Algorithm Mean (%)	80	37	79	38
Algorithm Standard Deviation (%)	3.1	2.6	4.2	2.3

Table 3. Results from the Forte Mattarello tests.

	Moore Farm	
Flight #	1	
# of Images	50	
Type of Overlap	End	Side
Flight Settings (App) (%)	70	70
Algorithm Mean (%)	54	61
Algorithm Standard Deviation (%)	10.7	5.7

Table 4. Results from the Moore Farm test

5. CASE STUDY: IMAGE FILTRATION

The filtration of the algorithm was tested on Flight 1 from the Nora dataset. It was processed a single-grid and as a double-grid (criss-cross) to compare the effects of filtration on the different block shapes. The flight had an end overlap of 80% and a side overlap of 40%. An end overlap of 60% and side overlap of 40% was used to test the ability of the algorithm to filter images. The images before and after the filtration were processed photogrammetrically. Figure 6 and 7 show the image block before and after filtration for the single-grid and double-grid configurations. The results of the filtration and processing of the block shapes are summarised in Table 5 and Table 6.



Figure 5. Irregular overlap clearly visible in three consecutive images of the same strip from the Nora dataset (Flight 1).

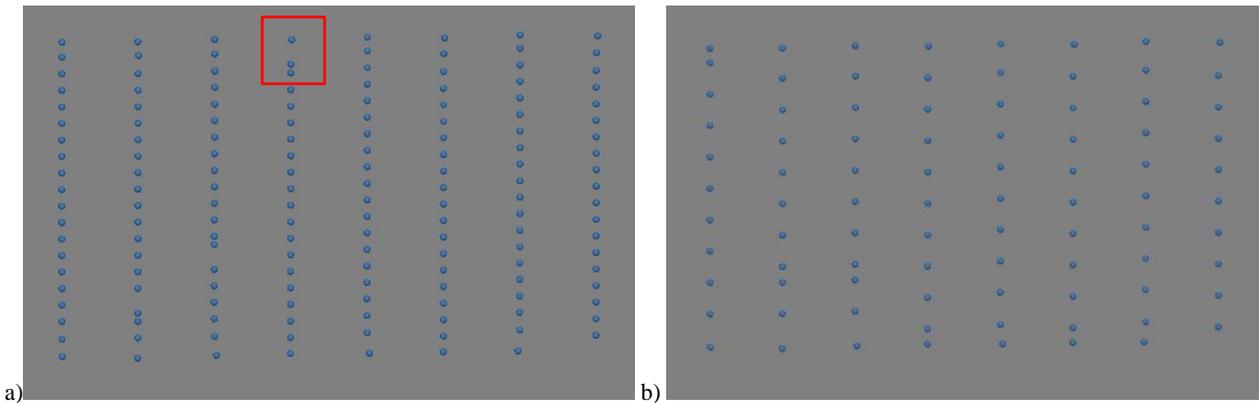


Figure 6. Image block of the single-grid flight before (a) after (b) filtration. An example of irregular overlap is shown.

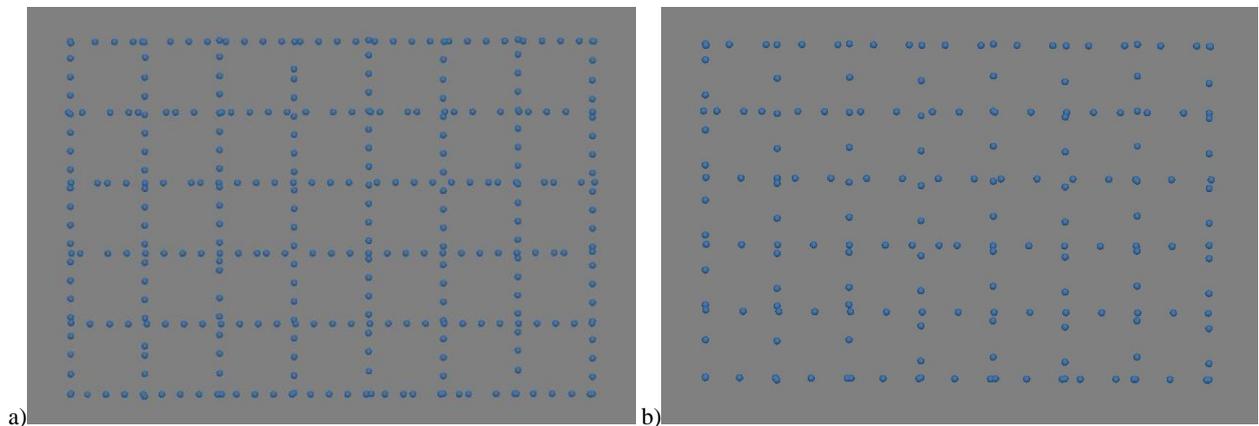


Figure 7. Image block of the double-grid flight before (a) and after (b) filtration.

	Before Filtration	After Filtration
# of Images	159	87
Bundle Adjustment Processing Time	40 minutes	13 minutes
Filtration Time	< 5 minutes	
Mean End Overlap	79	62
Mean Side Overlap	41	40
RMS Reprojection Error	0.54 px	0.56 px
# of 3D Points	126 626	118 162
# of Points in 2 images	32 744 (26%)	81 389 (69%)
# of Points in 3 images	28 029 (22%)	26 144 (22%)
# of Points in 4 images	25 972 (21%)	6 657 (6%)
# of Points in >4 images	39 881 (31%)	3 972 (3%)

Table 5. Results of the filtration test on the single grid dataset.

	Before Filtration	After Filtration
# of Images	334	185
Bundle Adjustment Processing Time	3 hours 17 minutes	54 minutes
Filtration Time	< 10 minutes	
Mean End Overlap	79	62
Mean Side Overlap	44	43
RMS Reprojection Error	0.78 px	0.75 px
# of 3D Points	130 039	141 799
# of Points in 2 images	25 348 (20%)	46 223 (33%)
# of Points in 3 images	15 056 (11%)	33 403 (24%)
# of Points in 4 images	12 942 (10%)	22 884 (16%)
# of Points in >4 images	76 693 (59%)	39 289 (27%)

Table 6. Results of the filtration test on the criss-cross dataset.

It can be seen in Figures 6 and 7 that the image filtration was successful. It reduced both UAV image sets by nearly half (as set in the preferences of the test), as expected, keeping the overlap constant as desired and reducing also the computational time. The single-grid test shows that though the 60% filtration leads to decreased processing times (33% of the time of the unfiltered data) it also leads to a large loss of redundancy, as expected. Though the filtration on the double-grid led to a similar reduction in processing time (27% of the time of the unfiltered data), led to redundancies that are much closer to the unfiltered data and even closer to the unfiltered single-grid data. The tests show that single-grid flights should not be subjected to significant image filtration whereas double-grid flights are more robust. It should be noted that the authors are not suggesting that 60% end overlap is sufficient, the number was just used to test the filtration. The Root Mean Square (RMS) reprojection error remained similar in all tests showing that the filtration procedure does not greatly affect it.

6. CONCLUSIONS AND FUTURE WORKS

This paper presented a methodology for analysing UAV datasets and filtering out unneeded images before photogrammetric processing. The implementation of this methodology is able to reduce block sizes and processing times while still ensuring that there is enough information to process the data in a reliable way. All these aspects are more and more mandatory, despite the increasing processing performances of software and computers. The loss of redundancy was due to the selected end/side overlap. Though an initial release of the algorithm is available, it is far from being complete. There are some limitations as the overlap calculated is only valid for relatively flat terrain and the reliability of the results is dependent on the Global Navigation Satellite System (GNSS) / Inertial Measurement Unit (IMU) used. The method is valid for flat terrain because the ground projection is calculated based on an elevation from the ground. As elements increase in elevation from that ground assumption, the overlap calculations will become farther and farther from the truth. The flat ground assumption makes the results from this methodology ideal for producing orthophotos or Digital Surface Models (DSMs).

The analysis and filtering have been developed only for strip/grid-based flights, thus the analysis of UAV flights flown in other patterns (circular or free) will be added. We will also combine the methodology with the detection of blur images. Manually checking blurry images in large datasets is time consuming and highly subjective. However, even the measurement of blur in images is non-absolute. The result is dependent on the images that an image is compared to. The blur detection Saturation Image Edge Difference Standard-deviation (SIEDS) algorithm is effective at detecting blur, however successful results are conditioned to images with a similar terrain as the process uses edge detection (Sieberth et al., 2016). The methodology discussed in this paper could be used to compare SIEDS values of overlapping images to help account for changes in terrain.

ACKNOWLEDGMENTS

The work was realized during an internship of A. Dhanda at FBK in Trento, Italy supported by the NSERC Create Heritage Engineering grant at Carleton University (Canada).

REFERENCES

- Barazzetti, L., Remondino, F., Scaioni, M., Brumana, R., 2010. Fully Automatic UAV Image-based Sensor Orientation. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 38, Part 1, Calgary, Canada.
- Bartelsen, J., Mayer, H., 2010. Orientation of image sequences acquired from UAVs and with GPS cameras. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 38.
- Benassi, F., Dall'Asta, E., Diotri, F., Forlani, G., Morra di Cella, U., Roncella, R., Santise, M., 2017. Testing accuracy and repeatability of UAV blocks oriented with GNSS-supported aerial triangulation. *Remote Sensing*, Vol. 9, 172.
- Colomina, I., Molina, P., 2014. Unmanned Aerial Systems for Photogrammetry and Remote Sensing: A Review. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 92, pp. 79-97.
- Eling, C., Klingbeil, L., Kuhlmann, H., 2015. A Direct Georeferencing System for the Real-Time Position and Attitude Determination of Lightweight UAVS. Proc. FIG Working Week, 1-17.
- Giordan, Hayakawa, H., Nex, F., Remondino, F., Tarolli, P., 2018. The use of remotely piloted aircraft systems (RPASs) for natural hazards monitoring and management. *Natural Hazards and Earth System Sciences*, Vol. 18.
- Gomez, C. and Purdie, H., 2016. UAV- based Photogrammetry and Geocomputing for Hazards and Disaster Risk Monitoring – A Review, *Geoenvironmental Disasters*, Vol. 3, pp. 1–11.
- Harvey, P., 2018. Exiftool by Phil Harvey, Exiftool. <https://www.sno.phy.queensu.ca/~phil/exiftool/> (Last accessed April 2018).
- Henderson, D. M., 1997. *Euler Angles, Quaternions, and Transformation Matrices – Working Relationships*. Mission Planning and Analysis Division NASA, Huston, Texas, USA.
- Horn, B. K.P., 1987. Closed-Form Solutions of Absolute Orientation Using Unit Quaternions. *Journal of the Optical Society of America*, Vol. 4(4), pp. 629-643.
- Lari, Z., El-Sheimy, N., Habib, A., 2015. System Considerations and Challenges in 3D Mapping and Modelling Using Low-Cost UAV Systems. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(3), p. 343.
- Luhmann, T., Robson, S., Kyle, S., Harley, I., 2011. *Close Range Photogrammetry: Principles, Techniques and Applications*. Whittle Publishing, Dunbeath, Scotland, p. 38.
- Nex, F., Remondino, F., 2014. UAV for 3D Mapping Applications: A Review, *Applied Geomatics*, Vol.6 (1), pp. 1-15.
- Pfeifer, N., Glira, P., Briese, C., 2012. Direct Georeferencing with On Board Navigation Components of Light Weight UAV Platforms. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 39(B7), Melbourne, Australia, pp. 487-492.
- Pix4D Support, n.d. Selecting the Image Acquisition Plan Type, <https://support.pix4d.com/hc/en-us/articles/202557459-Step-1->

[Before-Starting-a-Project-1-Designing-the-Image-Acquisition-Plan-a-Selecting-the-Image-Acquisition-Plan-Type#gsc.tab=0](#)
(Last accessed April 2018).

Rehak, M., Mabillard, R., Skaloud, J., 2013. A micro-uav with the capability of direct georeferencing. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 40(1/W2), pp. 317-323.

Sieberth, T., Wackrow, R., Chandler, J.H., 2016. Automatic Detection of Blurred Images in UAV Image Sets. *Journal of Archaeological Science*, Vol. 122, pp. 1-16.

Smith, R., Self, M., Cheeseman., 1986. Estimating Uncertain Spatial Relationships in Robotics. *Proceedings of the Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-86)*, pp. 267-288.

Tampubolon, W., Reinhardt, W., 2016. Direct georeferencing of UAV data based on simple building structures. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 41(B1), pp. 1043-1050.

Turner, D., Licieer, A., Wallace, L., 2013. Direct georeferencing of ultrahigh-resolution UAV imagery. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 52(5).

Wang, J., Garratt, M., Lambert, A., Wang, J.J., Han S. and Sinclair, D., 2008. Integration of GPS/INS/Vision Sensors to Navigate Unmanned Aerial Vehicles. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 37, Part B1-b, Beijing, China, pp. 963-969.

Whyte, L., Whyte, D., 1991. Simultaneous Map Building and Localization for an Autonomous Mobile Robot. *Intelligent Robots and Systems '91. Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop*, Vol. 3, pp. 1442-1447.