3D mapping and photogrammetry sensor payload for unmanned aerial vehicles

Kyla Purdon[1*], *Theo* van Niekerk[2], *Russel* Phillips[3] and *Stephen* Marais[1]

[1]Centre for Robotics and Future Production, CSIR, Pretoria, South Africa
[2]Mechatronics Department, Nelson Mandela University, Port Elizabeth, South Africa
[3]Mechanical Department, Nelson Mandela University, Port Elizabeth, South Africa

Abstract. Unmanned Aerial Vehicles (UAVs) have shown great potential for data collection and monitoring of areas. Sensors such as LiDARs and cameras can be used on UAVs for high-resolution data collection and used for various applications like Digital Surface Models (DSM), photogrammetry, inspection systems, and maintenance applications. This paper describes the design and implementation of a sensor payload for UAVs using a LiDAR sensor, camera, Inertial Measurement Unit (IMU), and Global Positioning System (GPS). The payload collects data that can be used to georeference LiDAR and camera data, which can later be used to generate a georeferenced map and perform object detection and classification.

1 Introduction

Unmanned Aerial Vehicles (UAVs) are also referred to as drones or aerial robots and are defined as a system that can navigate in 3D space without a human operator on board that can either be flown autonomously or controlled by a remote [1]. UAVs are useful data harvesting tools which can be used in environments that are difficult to work in when using traditional and time-consuming methods. The use of UAVs speeds up data collection, and enables work in potentially hazardous, and very difficult environments [2].

UAVs are used for a wide range of civil applications such as disaster management, construction and infrastructure inspection, agriculture and remote sensing, health care, waste management, utility inspecting, urban planning, wildlife conservation, geographic mapping, weather forecasting, mining, law enforcement, real-time monitoring of road traffic flow and commercial photography [1]. The four main uses are listed and described in Table 1:

Table 1: Main UAV applications and description of each application

Applications	Description and use case
Disaster management	Search and rescue operations to locate survivors. Situational awareness before the rescuers enter the area. Planning the rescue mission safely because they can plan for the exact scenario.

* Corresponding author: kpurdon@csir.co.za

Construction and infrastructure inspection	Help monitor project progress throughout the project. Used to map out the construction site. Can also be used to keep records of raw materials on site. Can be used to inspect buildings unsafe for human inspection.
Agriculture and remote sensing	Monitoring of crops and inspection of crops. Also aid in precision agriculture with crop disease inspection, nutrient deficiencies, and precision watering. Irrigation management with aerial analysis of the crops.
Health care	Delivery of emergency supplies and emergency equipment in hard-to-reach or rural locations. Blood and tissue sample collection from remote locations.

Unmarked pylons, masts, towers, cranes, or any tall structures can pose significant risks to a low-flying aircraft. Some potential risks include collision hazards, communication interference, navigation disturbance, and reduced visibility. To mitigate these risks, it is crucial to mark and properly identify pylons, mass, and towers with aviation warning lights, high-visibility paint, or other suitable markings. Additionally, implementing effective airspace management and conducting regular surveys to identify unmarked structures can help enhance aviation safety and prevent accidents caused by unmarked hazards. UAVs can be used to perform a cost-effective method to survey land that is being used by low-flying aircraft, identifying all hazards in the area.

The UAV would require a self-powered sensor payload that can generate an accurate map of the area and perform object detection of hazardous objects in the area and give accurate 3D coordinates of the object. Various research has been done on using different methods to gather data from aerial payloads. Qian, et al. [2] proposed a robust Simultaneous localization and mapping (SLAM) system that combines a camera and lidar data. The line and plane features were extracted from the LiDAR and used to compute the relative pose between consecutive frames in an Iterative Closest Point (ICP) based manner. Haala, et al. [3] collected both LiDAR and image data from one UAV and applied a hybrid georeferencing. This process integrates a photogrammetric bundle adjustment with direct georeferencing of the LiDAR point clouds. Bultmann, et al. [4] proposed a real-time semantic inference and a fusion of LiDAR scans and RGB images. The payload would thus require a LiDAR, camera, Inertial Measurement Unit (IMU), and a Global Positioning System (GPS) to be able to collect high-resolution data of the area, and to create georeferenced data for mapping and object detection.

The development of a UAV sensor payload for data collection is covered in this paper with the system being equipped with a Velodyne Puck Lite LiDAR sensor, a Sony a6400 camera, an Xsens MTi 680 IMU. A Global Navigation Satellite System/Inertial Navigation System (GNSS/INS) with Real-time Kinematic positioning is also included to increase the accuracy of the LiDAR and camera georeferencing.

This paper discusses in Section 2 the design of the payload, which is described in terms of architecture, mechanical, electrical, and software. Section 3 covers the data acquisition and processing of the data. In Section 4 the results of the data obtained from the camera, LiDAR, IMU, GPS, and the data fusion results are discussed. Finally, Section 5 contains the conclusion of the research.

2 Design

The design section explains the overall design of the payload to meet the requirements to collect high-resolution data from a LiDAR and camera, allowing for georeferencing of the

data. This is broken down into five subsections, namely Architectural Design, Sensors Selection, Mechanical Design, Electric Design, and Software Design.

2.1 Architectural Design

The design requirement is a self-powered payload that can be used for mapping and object detection which has a weight of less the 5 kg to allow for it to be mounted on the DJI M600. After the analysis from the literature review and above mentioned design requirements, the following sensors were identified as key elements in the UAV sensor payload: LiDAR, IMU, GNSS, and Camera. Each sensor was then specified to meet the high-resolution accuracy and weight requirements which is covered in more detail in Section 2.2, 2.3 and Section 3. The architecture was developed by understanding how the sensors interface with each other and using the communication methods of each of the sensors. Fig. **1**, shows the overall architecture of the sensor payload.

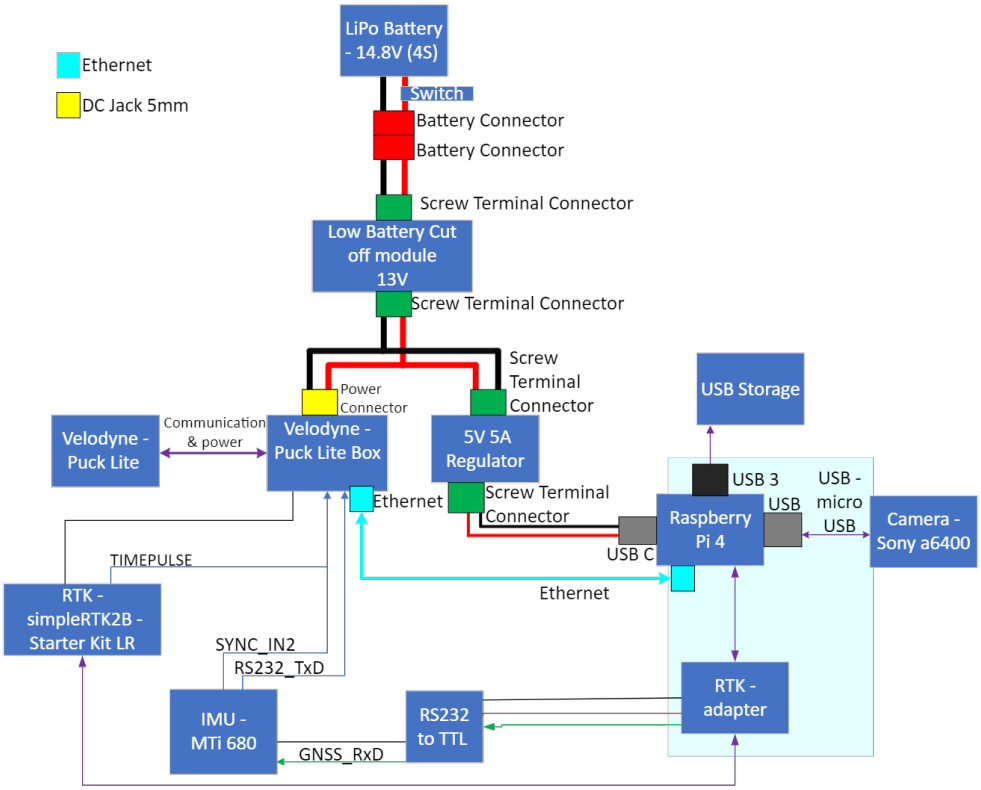

Fig. 1. Architecture of sensor payload.

To meet the requirements of the payload being a standalone system, an internal battery was incorporated into the design. This had the additional benefit of allowing the payload to be re-used by other mobile robotic platforms or vehicles. The processing unit to control the payload had to be small, lightweight, and have enough processing power to collect and store the data from the four mentioned sensors. For this a Raspberry Pi 4 acts as the control unit and is used to collect and store the data from the sensor payload. All post-processing of the data is performed externally.

2.2 Sensor Selection

The payload has four sensors, namely: LiDAR, camera, IMU, and global navigation satellite system/inertial navigation system. The camera is a Sony a6400 24.2-megapixel camera, a 23.5 x 15.6 mm Advanced Photo System type-C (APS-C) complementary metal-oxide semiconductor (CMOS) sensor with a focal length of 16 to 50 mm [5] and a weight of 519 g in the lens. The LiDAR is a Velodyne Puck Lite, a 16-beam laser with a horizontal field of view of 360˚ and a vertical field of view of 30˚with a ±3 cm range of accuracy, a max range of 100 m [6], and a weight of 780 g. The IMU is an Xsens MTi 680 that has 0.2˚ roll/pitch and can achieve centimeter-level position accuracy when integrated with an external RTK GNSS receiver [7]. The GPS is a simple RTK2B Pro with integrated RTK GNSS technology [8]. A labeled diagram of the sensor payload is shown in Fig. **2**.

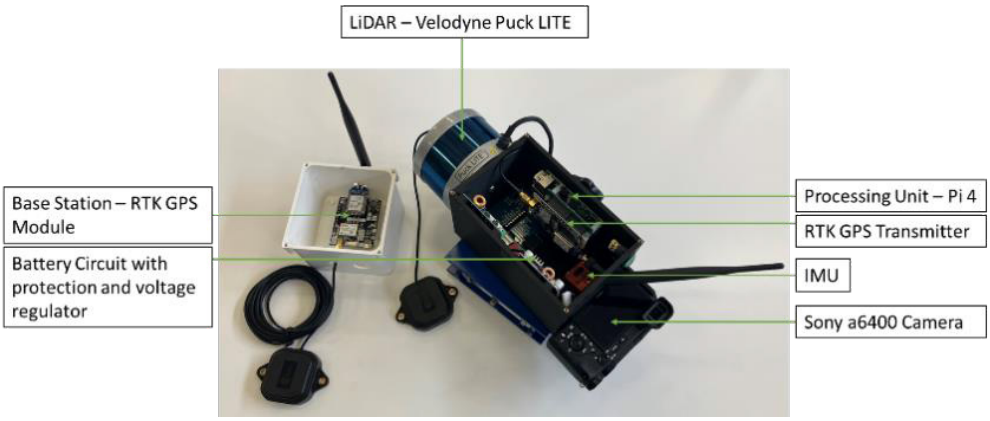

Fig. 2. Labeled sensor payload diagram.

2.3 Mechanical Design

The sensor payload is required to mount on a DJI M600 Pro [9], Fig. **3**, which has a carrying payload of 5.5 kg for 18 minutes using the TB48S battery. As part of the requirement, it was noted that it would be desirable if the payload could be used on the smaller DJI M300 which has a carrying payload of 2.7 kg. The design goal for the total mass of the payload is to be under 5 kg but aimed to be as close as possible to 2.5 kg to allow for use on the smaller M300. A 3D-printed enclosure was designed to minimize the weight and supporting structure for all the required components, with the payload coming out at a weight of 2.28 kg. The expected flight of the UAV with the designed sensor payload is 24 minutes using the TB48S battery. Note that this did not include the attachment frame which is required to mount the payload to the UAV.

Fig. 3. DJI M600 Pro

2.4 Electronic Design

The system consists of four sensors, a processing unit, this being the Raspberry Pi 4, and a battery with a battery protection system. To ensure accurate specification of the battery and voltage regulators, a thorough analysis was conducted to determine the voltage requirements of each sensor and processing unit. Table 2, shows the voltage and current requirements of each sensor and processing unit.

Table 2: Voltage requirements of each sensor payload component

Component	Voltage and current requirements
Processing unit	5 Vdc, 3 Ampere
LiDAR	9-18 V (approximately 1 A at 12 V)
IMU	4.5-24 V (approximately 200 mA at 5 V)
Camera	Own Battery
GPS	5 V, 80 mA

A voltage of at least 9 V is required, Lithium Polymer batteries operate in cells of 3.7 V, and a 4S battery of: 14.8 V would be adequate. The maximum current draw is around 5 A, and a maximum flight time of 24 minutes is required based on the maximum flight time the UAV can achieve with a 3 kg payload. The sourced battery to meet the requirements is a 3300 mAh 14.8 V (4S) battery. The payload battery should last for at least 30 minutes of flight time which is sufficient for the flight time of the UAV.

The sensors on the platform use different communication protocols and interface standards. The IMU uses the RS-232 interface to gather data from the GPS via the well-known National Marine Electronics Association (NMEA) protocol, but this device only has a UART TTL Interface, and hence an RS-232 to TTL converter is used. The same GPS unit also communicates directly to the Raspberry Pi computer via a Raspberry Pi adapter. Universal Serial Bus (USB) is used in communications between camera and processing unit where HTTP POST with JSON is used as protocol. The LiDAR uses Ethernet to communicate with the processing unit using the manufacturer's defined protocol which uses User Datagram Protocol (UDP) packets.

2.5 Software Design

The Raspberry Pi 4 is running Ubuntu 22.04 and Robot Operating System (ROS) 2 Humble. The data collection uses three ROS2 drivers to collect the GPS, IMU, and LiDAR data. The data is then post-processed and a ROS2 simultaneous localization and mapping (SLAM) package [10] is used to create a point cloud map in a LAZ file using the data fused from the sensors. The data is then further processed using Cloud Compare [11], QGIS (Quantum Geographic Information System) [12], and WebODM Drone mapping [13] software.

3 Data Acquisition and Pre-processing

The Data Acquisition and Pre-processing section explains the data obtained for each sensor: camera, LiDAR, IMU, GPS, and the integration of sensor fusion of the camera and LiDAR data. The experiment data was collected from a three-storey balcony, 8m above the ground.

3.1 Camera Application

The camera is attached to the payload and triggered by the processing unit to take photographs. The photos are required to have a 75% overlap for photogrammetry requirements [14]. To achieve a 75% overlap at a walking speed of 0.6 m/s the ground cover of each image would need to be calculated. To determine the camera's Field of View (FOV) the flying height of the UAV and camera specifications are required. Fig. **4**, is used to explain how the equations have been derived to determine the FOV and aerial coverage of the camera.

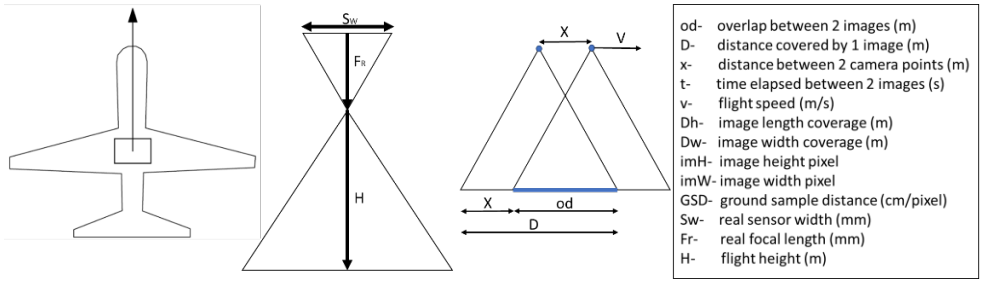

Fig. 4. Camera field of view to determine the ground coverage per photograph.

The key specifications of the camera used to determine the camera image ground cover from the UAV are shown in Table 3.

Table 3: Specifications of the camera to determine the ground coverage of the camera image.

Sensor type	23.5 x 15.6 mm (APS-C CMOS)
Sensor resolution	24.2 Megapixel (6000 x 4000)
Focal Length	16 to 50 mm (35 mm Equivalent Focal Length: 24 to 75 mm)

For this application, the payload is held above a three-story building 8m above the ground. Equations 1-6 [15] are used to determine the ground distance covered by one image and the ground sampling distance (cm/pixel).

To determine the image width coverage (Dw) the ratio of height (H), real focal length (Fr) and real sensor width (Sw) is used to solve for the unknown Dw.

$$\frac{H}{Fr} = \frac{Dw}{Sw} \tag{1}$$

The next step was to solve for the ground sampling distance (GSD) using Dw solved in equation 1. Once GSD is solved this is used to solve image length coverage (Dh) in equation 3.

$$D = D_w = \frac{imW \times GSD}{100} \tag{2}$$

$$Dh = \frac{imH \times GSD}{100} \tag{3}$$

To determine the overlap between two distances (od) the overlap percentage is multiplied by the image width coverage.

$$od = overlap \times D \tag{4}$$

To solve for the distance between two camera points (x) the overlap between two distances is subtracted from the image width coverage.

$$x = D - od \tag{5}$$

Finally, to determine the time interval between camera image shots the distance between two camera points is divided by the speed the UAV is traveling at.

$$t = \frac{x}{v} \tag{6}$$

The distance covered in the width direction is 7.83 m and the distance covered in the length direction is 11.75 m. The GSD is 0.20 cm/pixel, which means that each pixel represents 2 mm on the ground. In this test set up the assumed speed is 0.6 m/s and therefore the camera would be triggered every 3 s to achieve the 75% overlap, the frontal overlap covered is 5.86 m and the side overlap covered is 8.12 m.

3.2 LiDAR Application

The LiDAR is attached to the payload and the generation and storage of the point cloud is controlled by the processing unit. The LiDAR is attached to the UAV in the orientation of the beams being 90˚ to the flight direction and it is flown in the direction indicated in Fig. **5**. Equations 7-13 are derived from Fig. **5**, and are used to explain how the equations have been derived to determine the area of coverage of the LiDAR.

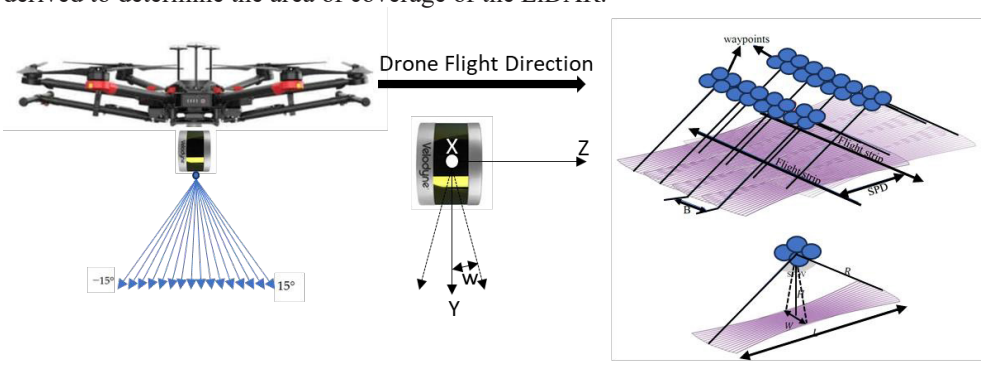

Fig. 5. LiDAR field of view to determine the ground cover.

The key specifications of the LiDAR to determine the point cloud ground cover from the UAV are shown in Table 4.

Table 4: Specifications of the LiDAR to determine the ground coverage of the point cloud.

Max Range	100 m
Horizontal FOV	360˚
Vertical FOV	30˚

Equations 7-13 [16] are used to determine the ground distance covered by the point cloud. To determine the width (W) coverage of the point cloud trigonometry is used to solve W because the height (H) is known and the vertical field of view of the LiDAR is also known. The length (L) of the point cloud is determined using Pythagoras.

$$W = 2tan\left(\frac{VFOV}{2}\right) \times H \tag{7}$$

$$L = 2\sqrt{(R^2 - H^2)} \tag{8}$$

The scanning field of view is also calculated using trigonometry.

$$SFOV = 2tan^{-1}\left(\frac{\frac{L}{2}}{H}\right) \tag{9}$$

The separation of distance between flight strips is calculated using the overlap percentage and the length of the laser scan, L.

$$SPD = (1 - sidelap\%) \times L \tag{10}$$

To determine the amount of flight strips required to cover an area, the width of the area is divided by the separation of the distance between flight strips plus an extra strip to count for both ends of the flight. A similar approach is followed to determine the number of waypoints per strip using the length of the area and the distance between points.

$$NFS = \frac{W_{area}}{SPD} + 1 \tag{11}$$

$$NScan = \frac{L_{area}}{B} + 1 \tag{12}$$

Then lastly the total way points is determined using the number of flight strips multiplied by the number of waypoints.

$$Total\ waypoints = NFS\ X\ NScan \tag{13}$$

The distance covered in the width direction is 4.29 m and the distance covered in the length direction is 200 m. In this test set up the assumed speed is a travel speed of 0.6m/s to achieve the 75% overlap.

3.3 GPS Application

The Global Positioning System (GPS) uses a rover and base configuration by using a data stream from the base station, to allow for the rover to output its relative position with cm-level accuracy in clear sky environments [8]. Fig. **6**, shows some of the data that can be collected from the GPS using U-center software.

Fig. 6. GPS data collected using U-center software.

3.4 IMU Application

The Inertial Measurement Unit (IMU) is a Global Navigation Satellite System/Inertial Navigation System (GNSS/INS) that can be integrated with an external Real-time kinematic (RTK) positioning system. This functionality allows for an improvement from meter-level

to centimeter-level accuracy [8]. Fig. **7**, shows a sample of the data that was collected from the IMU.

The IMU can be integrated with the GPS rover to increase the accuracy of the georeferenced data from meters to centimeters depending on the survey quality of the base station GPS data. In this study, we surveyed the data with a 0.5 m accuracy which took approximately 2 hours to survey-in. The GPS data is integrated into the cloud point to georeferenced the points. The data with ground points can also be georeferenced to the camera data.

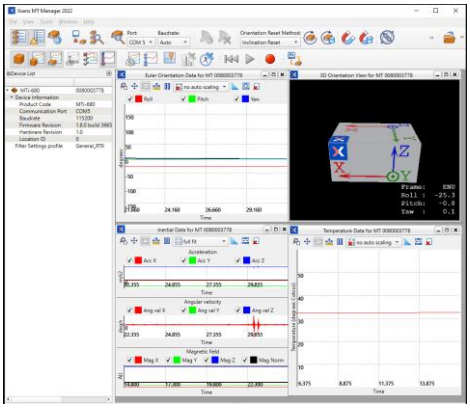

Fig. 7. IMU data collected using the MT Manager.

3.5 Integration of Camera and LiDAR data

The data overlap of the sensors is needed to ensure the sensors are seeing the same objects at the same time and sharing the same field of view. This aids in easier integration of the visualization of the camera and LiDAR data. The camera data needs to achieve a 75% overlap of data collection for mapping and photogrammetry requirements. LiDAR data needs to only achieve a 75% side overlap as the frontal overlap is constant as the UAV flies. The camera and LiDAR Fields of View (FOV) coverage overlap is shown in Fig. **8**.

From Fig. **8**, the camera covers 7.83 m in the length and the LiDAR covers 4.29 m in the length. However, this has no effect on the flight plan because the LiDAR frontal overlap can be ignored in the flight direction of the UAV. The height of the camera images covers 11.75 m and the LiDAR covers 200 m, this however does have an effect on the UAV flight plan. The camera needs to achieve the 75% overlap, and this will ensure the LiDAR overlap is actually 99% overlap.

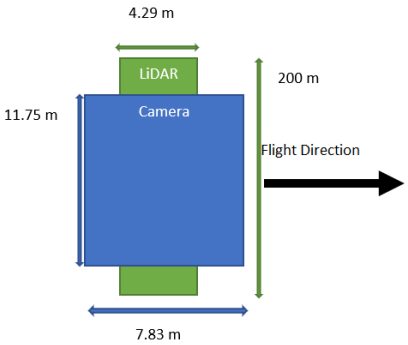

Fig. 8. Camera and LiDAR FOV together

4 Results

This section details various aspects of the collected data. In the first section the GPS position data is discussed without applied corrections versus the use of RTK-referenced data using the base and rover configuration. This is followed by a section on the results from the camera and LiDAR data, and the last section discusses the results of sensor fusion of all the data.

4.1 GPS data with and without corrections

The use of GPS with a rover and base station offers significant benefits in terms of precise positioning, real-time tracking, improved surveying efficiency, enhanced navigation, georeferencing, and synchronization. The GPS data without a base station operating as a single unit can achieve a 0.9 m accuracy whereas a GPS rover and base station configuration can achieve a 10 mm accuracy. The base station was surveyed with a 0.5 m accuracy which took approximately 2 hours. Fig. **9**, shows on the left the data obtained with no base station and on the right the data obtained with a base station.

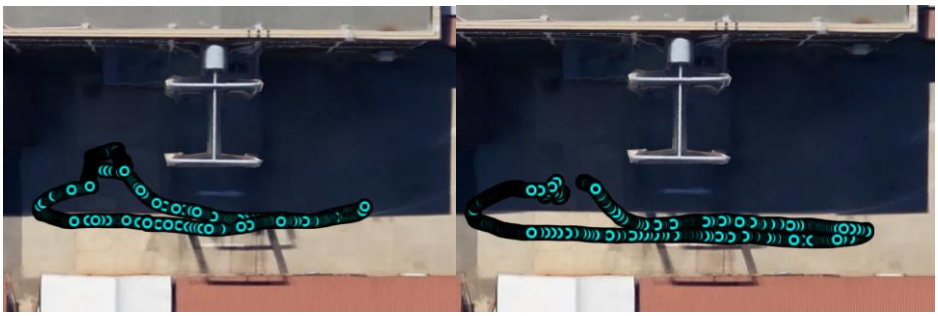

Fig. 9. Left, GPS data with no Base Station, Right, GPS data with Base Station

Looking at Fig. **9**, no clear effect of the base station can be seen. To show the effect of the base station two graphs of the covariance of the data collected with and without a base station are plotted in Fig. **10**.

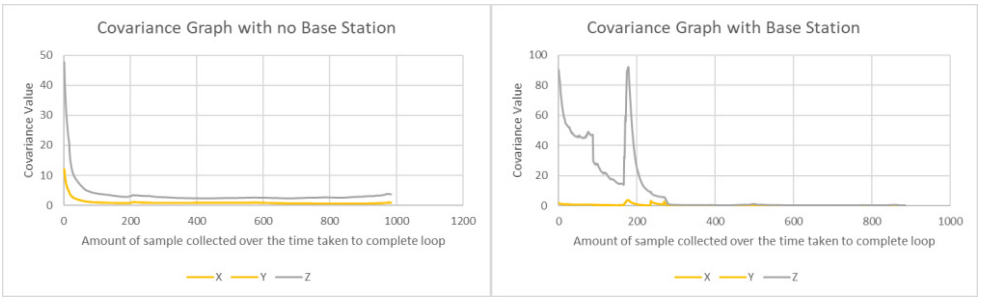

Fig. 10. Left, Covariance data without Base Station, Right, Covariance Data with Base Station

From the graphs in Fig. **10**, the following can be concluded. The GPS data with no base station achieved an average covariance value of 0.9. The GPS data with a base station takes some time to achieve the GPS fix, this is due to the z, altitude, value taking longer than the x, longitude, and y, latitude, to become stable. This is because of the arrangement of satellite triangulation configurations. The base station graph (right) shows initial spikes in the data where corrections were applied without a proper position fix. Once the position fix was

received at about 300 samples into the graph, the results become significantly better, where the average covariance of the GPS data with a base station is about 0.1.

4.2 Camera and LiDAR data

The data collected from the camera and LiDAR aids in the mapping and object detection of objects in the area under survey. It should be noted that the LiDAR data was obtained at ground level and not from an aerial view. The LiDAR data is not showing as clear details as the camera, but it aids in giving the 3D aspect which can be used with the camera data to accurately project the 3D bounding box of objects. In Fig. **11**, the image on the left is a point cloud generated by Cloud Compare [11], and on the right the point cloud is displayed in RVIZ2 [17], where the white line shows the path followed and recorded by the inertial measurement unit. RVIZ2 is a graphical interface that is a port of ROS2 used to display sensor data, robot information and maps.

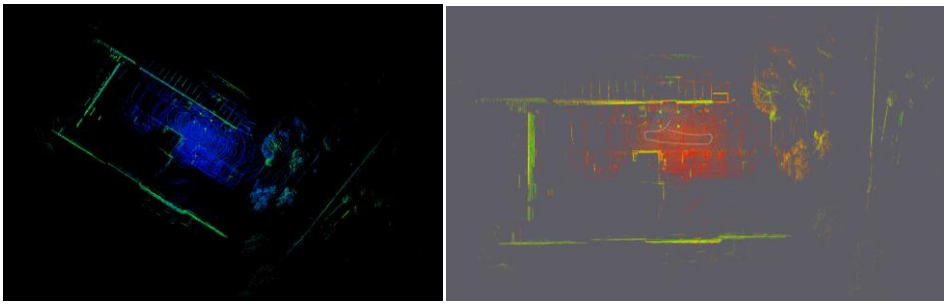

Fig. 11. Point cloud data on the left in Cloud Compare and the right in RVIZ2.

The image in Fig. **12**, is a Google Earth image of the point cloud obtained in Fig. **11**.

Fig. 12. Google Earth Arial image of point cloud data in Fig. **11**.

The image captured in Fig. **13**, was taken from an 8 m high building. The image shows a good photo quality image that can be used for object detection. Three pieces of paper are placed in an image, Fig. **13**, to verify the accuracy of the size of objects in the photo at a known height, real sensor width, and real focal length.

Fig. 13. Image captured with a camera at 8m height.

The size of each piece of paper was measured from the image and scaled according to the equations 1-6. The results of the dimensions of each piece of paper can be seen in Table 5.

Table 5: Size of objects in Fig. **13**

	Measured Size	Actual Size
A	154 mm	148.5 mm
	211 mm	210 mm
B	427 mm	420 mm
	294 mm	297 mm
C	214 mm	210 mm
	296 mm	297 mm

Analysis of the results obtained from Table 5, indicated that the size prediction is fairly accurate and can be used to determine the size of objects in the images.

The process of georeferencing the image data is currently a manual process using the QGIS (Quantum Geographic Information System) [12] open-source software. The coordinates from Google Maps are plotted on the image by selecting the points to match the image. Fig. **14**, shows the image on the right mapped onto the Google Maps image. In future work, the images will use ground control points and be automatically referenced in the software.

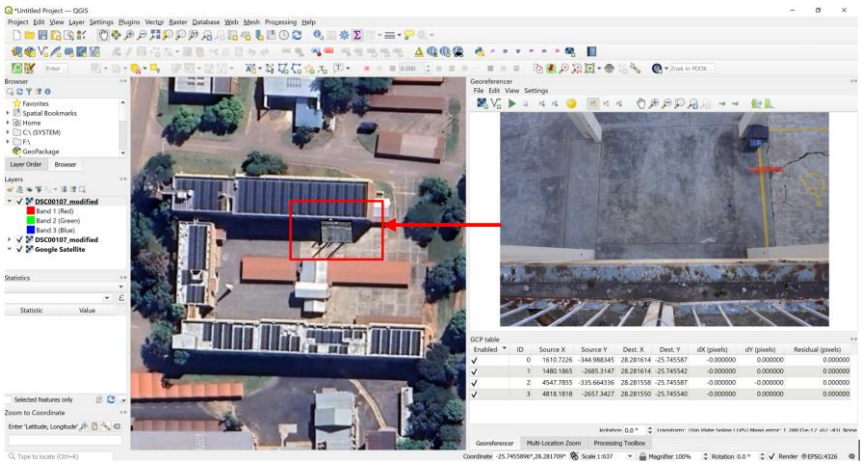

Fig. 14. Georeferencing Image Data using QGIS Software

To demonstrate the photogrammetry aspect of the small data set only 30 images were taken from the 8 m high balcony. The data was processed using WebODM Drone Mapping [13] open-source software. Fig. **15**, shows the 3D model built from the open-source software and compares it to the real image on the right.

Fig. 15. Photogrammetry on the left and actual image on the right

4.3 Fusion of data

Fusion of data has been shown to increase the accuracy and reliability of data [18]. For this paper, the focus was on creating a sensor payload that allows for sensor fusion of GPS, IMU, camera, and LiDAR data. Fig. **16**, shows the fusion of the IMU data and GPS data, together with the UCT date and time, and shows the integration of the IMU, GPS, and LiDAR data with the GPS position given at the bottom of the figure.

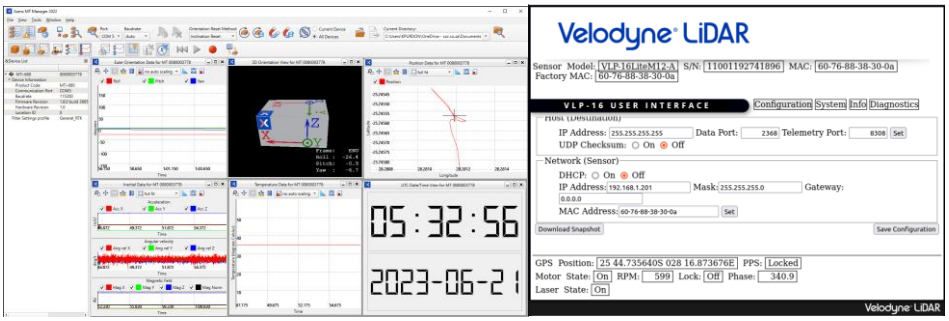

Fig. 16. Left: the fusion of GPS data with IMU, Right: GPS, and LiDAR data fusion.

The process of georeferencing the point cloud is a currently a manual process. In future work, the georeferencing of the point cloud would be added to the software to automate the process. In Fig. **17**, the left image shows the point cloud that was manually georeferenced in Cloud Compare and imported into the QGIS software to overlay it on the Google Earth maps. In the right image, the point cloud was coloured using an image from Google Earth and overlayed onto the georeferenced point cloud.

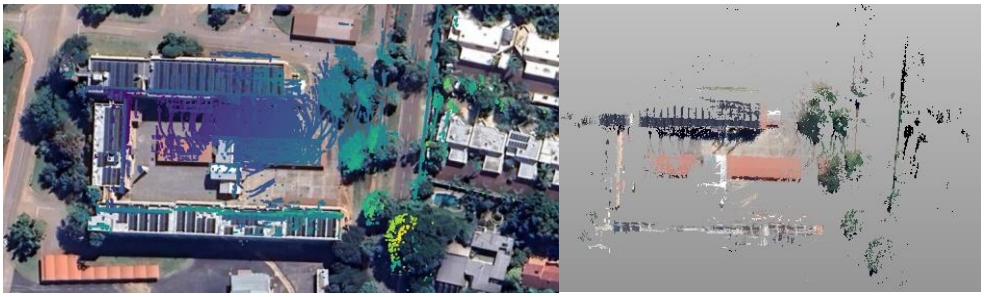

Fig. 17. Left, georeferenced point cloud overlaid on Google Earth and right, colour point cloud.

5 Conclusion

In conclusion, the payload which consists of GPS, IMU, camera, and LiDAR, can be used to collect reliable data from the environment. The GPS data was collected from the base and rover configuration and the results yield good accuracy once a fix from the base station was obtained. The IMU data was collected and integrated with the GPS and LiDAR data, to create georeferencing data. The generated camera images are of good quality and can be successfully used to generate photogrammetry data which can also be used for object detection and classification. Data collected from LiDAR displays an accurate point cloud of the environment.

Currently the georeferencing data is created manually using post-processing methods, and further investigations are taking place to in future achieve automatically georeferenced LiDAR results. To assist with camera georeferencing, use will be made of ground control points which enhance the accuracy of the position data.

It was shown that the payload was able to successfully collect data to create a georeferenced data set which can now be used for various applications like mapping, object detection, inspection, and or maintenance.

References

1. M. Sivakumar and N. Malleswari, "A Literature Survey of Unmanned Aerial Vehicle Usage for Civil Applications," *Journal od Aerospace Technology and Managment,* vol. 13, pp. 1-23, 2021.
2. J. Qian, K. Chen, Q. Chen, Y. Yang, J. Zhang and S. Chen, "Robust Visual-Lidar Simultaneous Localization and Mapping System for UAV," *IEEE Geoscience and Remote Sensing Letters,* vol. 19, no. 19, pp. 1-5, 2022.
3. N. Haala, M. Kolle, M. Cramer and D. Laupheimer, "Hybrid georeferencing of images and LiDAR data for UAV-based point cloud collection at millimetre accuracy," *ISPRS Open Journal of Photogrammetry and Remote Sensing,* vol. 4, pp. 1-11, 2022.
4. S. Bultmann, J. Quenzel and S. Behnke, "Real-Time Multi-Modal Semantic Fusion on Unmanned Aerial Vehicles," in *10th European Conference on Mobile Robots (ECMR)*, Bonn, 2021.
5. Sony, "Sony Support ILCE-6400 Specifications," Sony, 2021. [Online]. Available: https://www.sony.com/electronics/support/e-mount-body-ilce-6000-series/ilce-6400/specifications. [Accessed 11 May 2022].
6. Velodyne LiDAR, *Velodyne LiDAR Puck LITE Light Weight Real-time LiDAR Sensor,* 2019.
7. Xsens, *MTi 600-series Datasheet,* 2023.
8. "simpleRTK2B Pro," Ardusimple, 2023. [Online]. Available: https://www.ardusimple.com/product/simplertk2b-pro/. [Accessed 12 May 2022].
9. DJI, "MATRICE 600PRO SIMPLY PROFESSIONAL PERFORMANCE," DJI, 2023. [Online]. Available: https://www.dji.com/matrice600-pro. [Accessed 06 July 2022].
10. T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti and D. Rus, "Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping," *CoRR,* p. 2020.
11. CloudCompare, "CloudCompare," CloudCompare, [Online]. Available: https://www.cloudcompare.org/main.html. [Accessed 14 June 2022].
12. Development Team QGIS, "QGIS A Free and Open Source Geographic Information System," QGIS, 2009. [Online]. Available: https://www.qgis.org/en/site/. [Accessed 14 February 2022].
13. WebODM, "OpenDroneMap," OpenDroneMap contributors, [Online]. Available: https://www.opendronemap.org/webodm/. [Accessed 14 June 2022].
14. "How to verify that there is enough overlap between the images - PIX4Dmapper," PIX4D, 2011-2021. [Online]. Available: https://support.pix4d.com/hc/en-us/articles/203756125-How-to-verify-that-there-is-enough-overlap-between-the-images-PIX4Dmapper. [Accessed 11 May 2022].
15. "Step 1. Before Starting a Project > 1. Designing the Image Acquisition Plan - PIX4Dmapper," PIX4D, 2011-2021. [Online]. Available: https://support.pix4d.com/hc/en-us/articles/202557409-Step-1-Before-Starting-a-Project-1-Designing-the-Image-Acquisition-Plan-PIX4Dmapper. [Accessed 11 May 2022].
16. A. H. Lassiter, T. Whitley, B. Wilkinson and A. Abd-Elrahman, "Scan Pattern Characterization of Velodyne VLP-16 Lidar Sensor for UAS Laser Scanning," *Sensors(Basel),* vol. 20, no. 24, p. 7351, December 2020.

17. RVIZ2, “rviz repository,” 16 March 2023. [Online]. Available: https://index.ros.org/r/rviz/. [Accessed 14 April 2023].
18. G. A. Kumar, J.-H. Lee, J. Hwang, J. Park, S.-h. Youn and S. Kwon, “LiDAR and Camera Fusion Approach for Object Distance Estimation in Self-Driving Vehicles,” *Symmetry,* vol. 12, no. 2, p. 324, February 2020.