IMAV2017-41
https://www.imavs.org/pdf/imav.2017.41



# An Automated Rapid Mapping Solution Based on ORB SLAM 2 and Agisoft Photoscan API

Markus Bobbe*, Alexander Kern, Yogesh Khedar, Simon Batzdorfer and Ulf Bestmann
TU Braunschweig, Germany

ABSTRACT

This paper describes a system consisting of an UAV and a ground station capable of automated mapping based on aerial images. The focus of the presented system is to obtain georeferenced orthophotos within a short time frame.

Two approaches have been implemented in the system: an online visual SLAM based on ORB SLAM 2 and a photogrammetry pipeline using the Agisoft Photoscan API. Both approaches will be described and its result evaluated and compared.

## 1 INTRODUCTION

After natural disasters, responding forces rely on accurate maps to apply their ressources as fast and efficient as possible. Available maps and satellite images are often outdated or rendered useless due to the disaster (e.g. floods, earthquakes). UAVs can be used by first response forces to generate up to date georeferenced orthophotos. Aerial photogrammetry is a proven tool for the creation of such data. In recent years UAVs are used increasingly to generate the required images. Specialised software also simplified the photogrammetry workflow. Yet this workflow still requires knowledge regarding flight planning, data handling and processing parameter selection. The typical use case is the processing of survey data which is rarely time sensitive. The average processing therefore takes several hours since quality is often the most important criteria. Investigations regarding the best quality with minimal processing time are rare.

Another approach is the use of visual SLAM solutions which are capable of augmenting the solution incrementally with every new image. Available algorithms, often used in robotic reasearch, are less straightforward to use and have to be adapted for the specific use case. The results usually don't have the quality produced by photogrammetry solutions and investigations usually don't focus on the accuracy. Both approaches were adapted and implemented with the goal to gain a full automated rapid aerial mapping solution. This includes the mission planning, camera control, image transport to ground station, automated processing and the visualization of the results. The live mapping (SLAM) approach is based on the ORB-SLAM algorithm and updates the map when a new image is available. The photogrammetry based approach uses the commercial photogrammetry software Agisoft Photoscan. The processing is started with predifined parameters using the Photoscan API once the mapping mission is completed. In this paper the hardware and software setup for the developed system will be described and the generated results from both approaches will be compared.

In section 2 the hardware setup used for acquiring the images on-board the UAV and transferring them to the ground station is given. The implemented ROS based software infrastructure is also described. In section 3 the ORB SLAM 2 based solution is presented. The solution based on the Agisoft Photoscan API is given in section 4. Section 5 presents the mapping results generated by both solutions during flight tests and compares them regarding quality and speed.

Figure 1: Air Robot AR200.

## 2 SETUP

The setup is integrated in a modified Air Robot AR200 hexacopter displayed in figure 1. It carries a payload of 2.7kg leading to a maximum flight time of 25min. The payload consists of an Air Robot 2-axis camera gimbal (figure 2), an Intel Nuc i5 on-board PC, a experimental navigation package (consisting of 2 Analog Devices ADIS16488A IMUs, a uBlox M8T GNSS receiver and a Phytec Mira Cortex A9 processing board), a Gateworks GW 5520 wifi board and a AVT Manta 917G GigE Camera with a Cinegon 1.9/10mm lens. The ground station is equipped with the same wifi board and is based on an Intel i7 CPU and a GeForce GTX1080 GPU.

The systems uses ROS as the underlying framework. Figure 3 displays a flowchart of the implemented workflow. The

*contact: m.bobbe@tu-braunschweig.de

**IMAV2017-41**
https://www.imavs.org/pdf/imav.2017.41

Figure 2: Gimbal with Camera and IMU.

sensor data is received by dedicated nodes on the aerial vehicle. The camera pose information is added to the image in the *Geo Image Flight Node*. The image is then sent to the *Geo Image Ground Node* on the Ground Station. The SLAM process is separated in the *SLAM Tracker Node*, which calculates the tranformation between images and the *Stitcher Node* which applies the transformations. The transformed images are displayed by the *Visualization Node*. The *Photogrammetry Node* receives the georeferenced images and triggers the photogrammetric processing ones the survey is finished. The results are also displayed in the *Visualization Node*.

## 3 SLAM Process

During the past few years computer vision especially SLAM based algorithms have undergone rapid development. Klein and Murray showed in their highly regarded work [1] how a novel pose estimation only by monocular image processing can look like and what great potential comes with it not only for augmented reality but the whole robotic scene. Integrating bundle adjustment and splitting up tracking and mapping into seperate threads were followed by a strong, realtime capable framework that had a significant impact on visual SLAM. However their approach lacked several, essential properties a modern SLAM needs for applications outside the academic use. Especially robust loop closure and relocalization was difficult to integrate into their framework, as they utilized separate features for tracking and place recognition. Mur-Artal and Montiel took up the basic principles proposed for PTAM and developed efficient solutions for these challenges. Their work led to the new designed ORB SLAM 2 [2] that reaches unprecedented accuracies and published it open source. It is briefly described in section 3.1.

By building up on that framework, we propose a pipeline to take advantage of the accurate camera pose estimated by the SLAM to generate large 2D aerial maps in realtime in section 3.2, similiar to those coming from modern photogrammetry software. A further comparison between these two approaches is then evaluated in section 5.

### *3.1 ORB SLAM 2 framework*

The general structure of the ORB SLAM 2 framework is displayed in figure 4. Gray underlaid boxes represent seperate threads, while in the middle also main components of the implemented map and place recognition are shown.

**Tracking**
After initialization, which is explained in detail in [2], the tracking thread uses a constant velocity model to predict the current pose from the latest known position and movement. With this rough estimation a further analysis of features in the current region of interest is carried out. If the tracking state is triggered ”lost”, relocalization in the global map starts. After the tracking step a temporary pose is published and created keyframes are passed to the local mapping.

**Local Mapping**
In the local mapping thread a new identified keyframe is inserted as a node into a covisibility graph structure, which contains all relevant informations and relations as edges between the nodes. To achieve high accuracies after the tracking a local bundle adjustment is also carried out in this step.

**Loop Closing**
One of the main improvements of ORB SLAM 2, compared to PTAM introduced before, is the usage of only one type of features for all tasks within the framework. By taking advantage of a *bag of words approach* this allows the implementation of a place recognition to find loops in the graph and restructure it if neccessary. Based on that graph a global bundle adjustment is performed as soon as a loop is detected. In consequence a global consistent solution is achieved, even though the estimation error drifted over time.

### *3.2 Ortophoto Pipeline*

After estimating the camera pose using ORB SLAM 2 framework in the next step these informations are used to gradually create a 2D orthophoto of the area during flight. To achieve this a lightweight image projection combined with georeferencing based on similiar transformation is proposed in this section.

#### 3.2.1 Projection model

The projection model used in this paper follows the standard pinhole camera [3]. Therefore a world point $\underline{X} = (x, y, z)^T$ is described as

$$\underline{X} = s(\underline{\underline{RK}}^1 \underline{x}) + \underline{t} \tag{1}$$

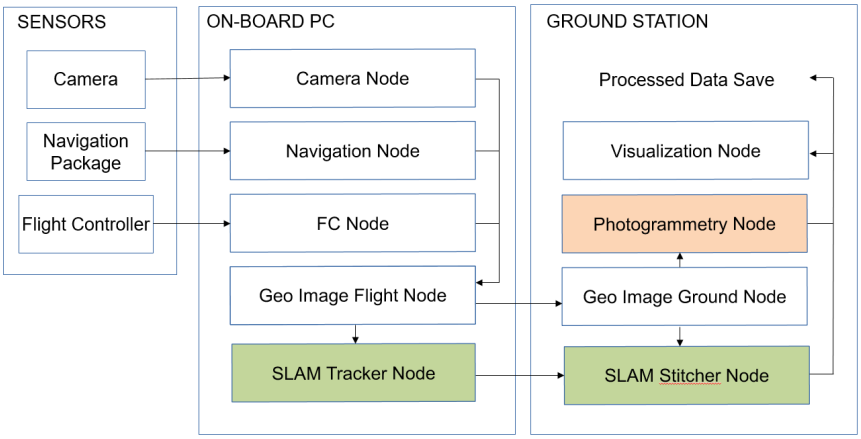


Figure 3: ROS node layout with SLAM (green) and Photogrammetry workflow (red).

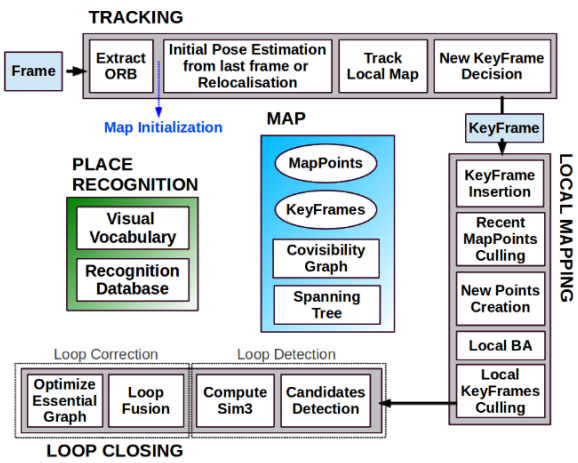


Figure 4: ORB SLAM 2 framework structure [2].

where $\underline{\underline{R}}$ is the 3x3 camera rotation in the world frame, $\underline{\underline{K}}$ the 3x3 camera intrinsic calibration matrix, s the scale factor of the projection direction vector, $\underline{x} = (u, v, 1)^T$ a point in the image plane in homographic coordinates and $\underline{t} = (t_x, t_y, t_z)$ the exterior position of the camera. The model is further extended by additional parameters $k_1$, $k_2$, $p_1$, $p_2$ and $k_3$ to compensate geometric distortion caused by the lens. Usually image points are therefore undistorted first by following the equations

$$x_{corrected} = x + [2p_1xy + p_2(r^2 + 2x^2)] \quad (2)$$

$$y_{corrected} = y + [p_1(r^2 + 2y^2) + 2p_2xy] \quad (3)$$

and projected applying the proposed formulation in 1 afterwards.

### 3.2.2 Image Projection

To create a 2D orthophoto from 3D camera poses a definition of the actual common reference plane is needed. The reference plane must be orthogonal to all camera observations to ideally reduce perspective errors to zero. Even when using gimbal stabilized data and a low distortion lens this error can not be completely avoided. Therefore an orthogonal reference plane can only be approximated. In this implementation this problem was solved by collecting a minimum amount of measurements in a buffer and calculating the best fitting solution of the sparse pointcloud generated by ORB SLAM 2. This approach is only valid as long as the ground is flat and the flight altitude relatively high. Applying additional RANSAC formulation reduces further influence of outliers. As soon as a valid reference frame is estimated for the first measurements, it is kept fixed for the rest of the series. Therefore large misalignements in the beginning can not be compensated afterwards. In the future it might be beneficial to check validity of the reference plane periodically and recalculate the current map. In the next step image boundaries can be projected into the reference plane by using the formulation of section 3.2.1. With these four points an exact perspective transformation with 8 DOF can be calculated. It is then further applied on the whole image and stitched to the current global map.

### 3.2.3 Geo-referencing

Simultaneously with the calculation of the common reference plane an additional geo-referencing for the images is estimated. It allows to transform every pixel of the final map to a defined lat/lon-coordinate. To achieve this, all images are geotagged in the moment they are captured with the latest available GNSS informations. After choosing the first complete measurement consisting of visual pose and global

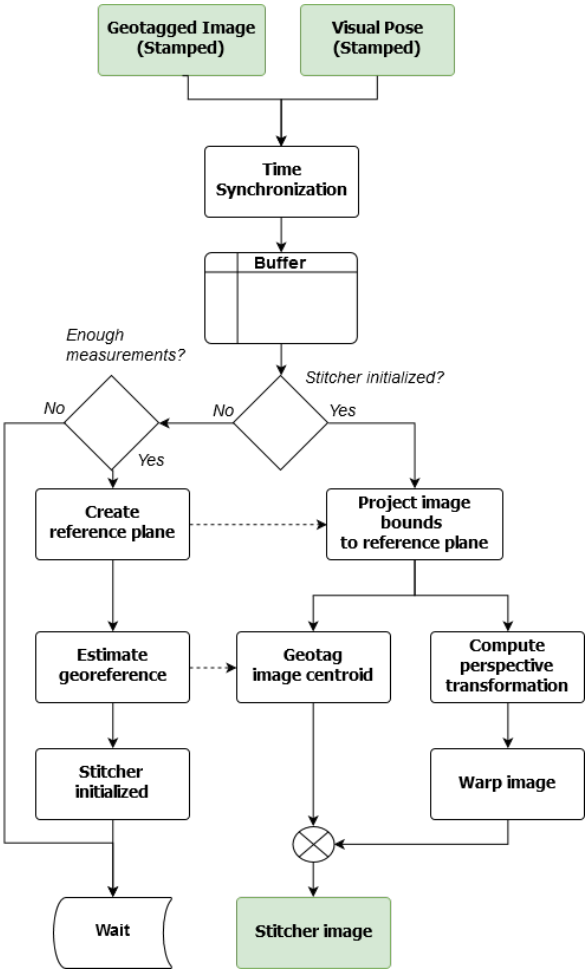


Figure 5: Structure of the orthophoto stitching.

GNSS position each as root of a local coordinate system, the following steps are performed for every image in the initial buffer:

**1)** Calculate image centroid in common reference plane

**2)** Transform centroid to local coordinate system (relative to centroid of first measurement)

**3)** Convert GNSS position from WGS84 to UTM32

**4)** Transform GNSS position to local coordinate system (relative to GNSS position of first measurement)]

Afterwards a 4 DOF similiar transformation (x, y, rotation, unitary scale) between these two local coordinate systems can be calculated by linear programming. Considering the root of the local coordinate systems a transformation to the global world is also possible.
While this approach is straighforward, it induces several uncertainties. By projecting the centroids in the reference plane and assuming their plane position matches the global position measured in the air, an error linked to the stabilization quality of the gimbal can be expected. Depending on the update rate of the GNSS receiver and the FPS of the camera different images can be tagged with the same position leading to a bad initialization. Therefore only the first appearence of every global position is considered in the described process. However asynchronous time between geotagging and image capturing results in an additional error depending on the speed of the UAV. At last by converting WGS84- to UTM32-coordinates affine transformation during linear programming can be avoided. This allows more intuitive handling of the coordinates, but also applies an uncertainty of several centimeters depending on the size of the map.

## 4 Photogrammetry Process

The photogrammetry process was implemented using Agisoft Photoscan 1.2.6. Extensive studies regarding the accuracy attainable by the software have been done before, for instance by Gini et al. [4] and Ouedraogo et al. [5]. Yet studies which focus on the minimal processing time are not available. In this implementation the Photogrammetry Node saves images belonging to the current mission and loads them into Agisoft Photoscan using the API once the survey mission is finished. The camera location and the inner camera geometry were written to the EXIF file of each image by the *Geo Image Ground Node* to enable its usage to georeferencing the results. The process is started using preselected processing parameters. The parameters affect the required processing time and the quality of the result.

The process consists of 4 steps. During the camera alignment features are identified and compared to optimize the homographic equation and determine the camera locations. In the next step a mesh is generated using the generated tie points. The mesh is then used to create an orthophoto and to export it in the desired format in the last step. It is possible to use a dense cloud or a Digital Elevation Model as the basis for the orthophoto generation. Yet the creation of these models would significantly increase processing time and was therefore rejected.

To determine a reasonable compromise between quality and processing time, 4 profiles with different quality parameters were created and and tested in the next section. The relevant parameters of each profile are given in table 1.

| profile | alignment accuracy | mesh face count |
|---|---|---|
| Agi lowest | lowest | low |
| Agi low | low | low |
| Agi medium | medium | medium |
| Agi high | high | high |

Table 1: Photogrammetry parameter profile defentition.

IMAV2017-41
https://www.imavs.org/pdf/imav.2017.41



## 5 Evaluation

To demonstrate the implemented workflow and to validate and compare the created results the complete system was tested by flying a sample mission. To estimate the accuracy of the created maps, 8 ground reference points (GRPs) were distributed over the mission area. The location of the GRPs were determined using a RTK-GNSS system leading to a horizontal RMSE below 2cm. To enable robust processing the overlap and the sidelap was chosen to be 70%. This resulted in a creeping line mission consisted of 4 times 130m lines with a distance of 25m and an altitude of 100m over ground.

The image rate and therefore the frontlap varies between the two implementations. The SLAM node receives the images with a higher framerate which is benefical for the tracker. The processing power of the earlier described ground station enables processing with a frame rate of up to 4Hz. This leads to an frontlap ov 99% and 864 images, yet only 27 were used as keyframes. The photogrammetry node receives the images with a lower framerate to limit the total number of images to minimize processing time. The chosen framerate of 0.3Hz leads to an overlap of 85%. In total the photogrammetry node received 47 images in less then 3 minutes flight time.

The photogrammetry pipeline was successfully tested in the field. To compare the presented profiles, they were triggered one after another. The created results are displayed in figure 8. All profiles provided consistent solutions and were succesfully georeferenced. The visual comparison gives no significant differences. The remaining profiles let to similar results also without notable differences to visual inspection. The processing time varied between 0.9 and 3.2 minutes. A comparison of this and other criteria is given in figure 6. The calculated position of the GRP was determined in each orthophoto. The derivation to the reference measurement is given in table 2. While the mean error in the low profile is two thirds of the error in the lowest profile, the calculated errors using the medium and high profiles are not enhancing significantly. The number of tie points created by the lowest profile is an order of magnitude lower compared to the other three profiles. The reprojection error roughly halves with each profile step. The xy camera error describes the mean horizontal camera displacement regarding the inital GNSS location and the final calculated position and is almost identical in all profiles.

To accelerate the photogrammetry process, the resolution of the created orthophoto can be reduced. Figure 9 displays the total processing time using the lowest profile with different target ground sampling distances (GSD) during orthophoto creation. The processing time can be reduced significantly and reaches processing times of 7 seconds when a orthophoto with an GSD of 0.4m is created.

The created final image of the SLAM pipeline is displayed in figure 8. The image was updated with every new image and was therefor finished before the copter landed. It reaches a GSD of 0.05 m, while the mean location error mea-

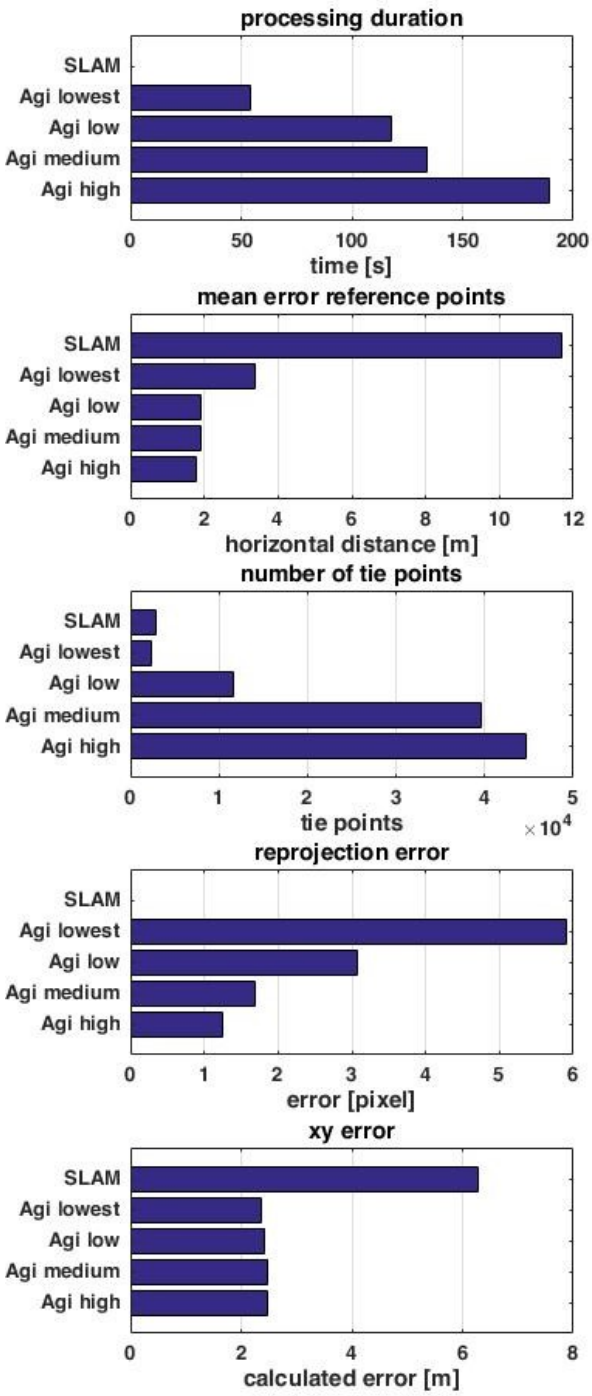


Figure 6: Mapping evaluation results.

| GRP | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | mean |
|---|---|---|---|---|---|---|---|---|---|
| SLAM | 15.83 | 18.94 | 14.56 | 16.57 | 9.21 | 10.15 | 4.65 | 4.73 | 11.67 |
| Agi lowest | 2.49 | 2.19 | 3.40 | 3.17 | 4.03 | 3.80 | 4.58 | 3.66 | 3.39 |
| Agi low | 1.41 | 1.28 | 1.87 | 1.74 | 2.19 | 2.06 | 2.49 | 2.46 | 1.91 |
| Agi medium | 1.34 | 1.13 | 1.80 | 1.70 | 2.15 | 2.13 | 2.64 | 2.63 | 1.91 |
| Agi high | 1.17 | 1.08 | 1.64 | 1.65 | 1.94 | 1.99 | 2.38 | 2.41 | 1.77 |

Table 2: Vertical error at reference points in meter.

Figure 7: Orthophotos created with the profiles lowest and high

Figure 8: Orthophotos created with SLAM approach

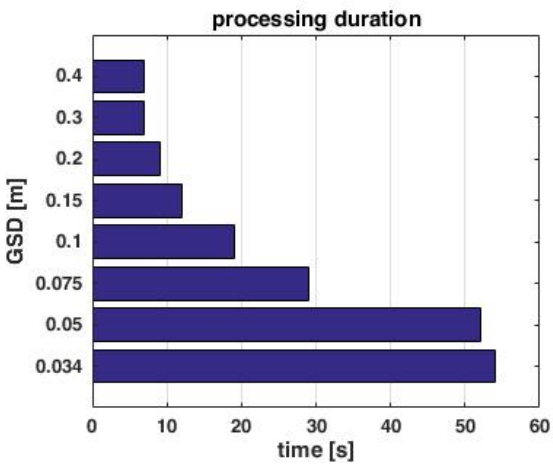


Figure 9: Processing Times using the Agi lowest profile and different GSD.

**IMAV2017-41**
https://www.imavs.org/pdf/imav.2017.41



sured using the reference points was about 12m which is significantly larger than the errors observed in the photogrammetry results. This and other results are also given in figure 6.

## 6 Conclusion

The introduced system demonstrated the proposed capabilities regarding automated camera operation, image transport and ortophoto generation. Both implemented mapping pipelines generated adequate results which fullfilled the requirements in a disaster scenario. The SLAM approach is capable of delivering instant results but the overall georeferencing accuracy is roughly one order of magnitude lower compared to the photogrammetry pipeline. The comparison of the different photogrammetry profiles indicated that the usage of the second fastest profile low is recommended for most scenarios since it is the fastest profile that is capable of reconstructing the complete area. Also the more elaborate profiles do not lead to significantly better results regarding georeferencing accuracy.

## Acknowledgements

The presented work was done within the joint research project ANKommEn funded by the German Federal Ministry of Economic Affairs and Energy administrated by the Space Administration of the DLR (funding code: 50NA1518).

Federal Ministry
for Economic Affairs
and Energy

## References

[1] Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

[2] Mur-Artal Raúl Montiel, J. M. M. and Juan D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[4] Daniele Passonib Livio Pintob Giovanna Sonaa Paolo Dossoc Rossana Ginia, Diana Pagliarib. UAV Photogrammetry: Block triangulation comparison. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-1(W2), 2013.

[5] Charles Debouche Jonathan Lisein Mohamar Moussa Oudraogo, Aurore Degr. The evaluation of unmanned aerial system-based photogrammetry and terrestrial laser scanning to generate dems of agricultural watersheds. *Geomorphology*, 214:339–355, 2014.