Advancements on Semantic Real-Time UAV Mapping

Marc Haubenstock[+*], Jules Salzinger[+], Michael Schwingshackl, Felix Bruckmüller, Christoph Sulzbachner, Phillipp Fanta-Jende

Austrian Institute of Technology - Center for Vision, Automation and Control
Email: (marc.haubenstock, jules.salzinger, michael.schwingshackl, felix.bruckmueller, christoph.sulzbachner, phillipp.fanta-jende)@ait.ac.at

Keywords: Digital Surface Model, UAV, Crisis and Disaster Management, Mapping, Real-Time UAV Mapping, Semantic Segmentation

Abstract

This paper presents key improvements in real-time ortho image generation and scene understanding for disaster management and first responders. Through the introduction of an Inertial Measurement Unit, a depth estimation network and a trained network for scene segmentation, it is possible to produce end-to-end real-time ortho and semantic maps. Since datasets containing inertial data are sparse, the results of the pipeline were verified on a flight, which was recorded and post-processed as a ground truth with ground control points using the standard photogrammetric workflow. The reported errors are in the same range as a post-processed ortho map on raw Global Navigation Satellite System measurements, however, produced in real time. Semantic segmentation results demonstrate surprising levels of accuracy and robustness, but reveal a need for more comprehensive data acquisitions and benchmarks.

1. Introduction

Unmanned Aerial Vehicles (UAVs) have gained a lot of traction throughout the last decade in the fields of rapid mapping and Digital Surface Model (DSM) construction (Hermann et al., 2024). In particular, in disaster response scenarios where accurate and real-time feedback is critical, UAVs have been the subject of research due to their high-resolution imaging systems and flexible usage (Erdelj et al., 2017).

This work builds on previous attempts for real-time UAV semantic mapping (Fanta-Jende et al., 2023). This workflow entails realtime generation of orthorectified tiles, DSM generation and semantic segmentation while only using raw Global Navigation Satellite System (GNSS) measurements and no global optimisation such as Bundle Adjustment. It reports improvements in state and depth estimation, as well as scene segmentation. Specifically, it improves the alignment of the estimated Simultaneous Localisation and Mapping (SLAM) trajectory with the GNSS measurements, by integrating an Inertial Measurement Unit (IMU) sensor, and the quality of the depth reconstructions through the usage of Cascaded View Aggregation (CVA). Furthermore, it leverages recent foundation models and several aggregated datasets to provide semantic maps of improved accuracy for first responders. In doing so, gaps in data quality and quantity are identified in the crisis and disaster management domain.

2. Related Works

2.1 Real-time aerial mosaicing

Early works used image based, 2D homographies to provide a larger compose image of multiple aerial views (Botterill et al., 2010). However, neither the state of the UAV nor 3D structure of the environment were recovered. Following this Bu et al. used monocular SLAM to estimate the state of the aerial UAV (Bu et al., 2016). However, scene reconstruction was still limited to planar geometry and therefore provided a simplified reconstruction of the observed scene.

Current state-of-the-art ortho mosaicing methods use a a sparse 3D representation of the imaged terrain which can be incrementally built and optimised as to reject outliers and provide a clearer ortho map. Different flavours of this pipeline exist which differ in what sensors are integrated at what stage of the optimisation pipeline. (Wang et al., 2019, Kern et al., 2020), use GNSS information as a loose fusion modality, such that after the bundle adjustment of the 3D to 2D correspondences has been completed, the estimated state trajectory is aligned with the GNSS information. (Zhao et al., 2022) uses a tight fusion, where the GNSS error term is optimised directly in the non-linear cost function. This has the benefit that the system state is jointly optimised for image and GNSS modalities concurrently, which results in a better estimate. Furthermore, when parts of the sensor information is missing, it may still be possible to perform a trajectory estimate which increases the system's robustness. However, very few works focus on integrating IMU data in addition to image and GNSS information. One reason may be the lack of public datasets with recorded IMU data (Montgomery et al., 2021, Rahnemoonfar et al., 2023). (Liu et al., 2025) uses IMU data, however it is only used to augment the estimate of the UAV's rotation and is not tightly integrated into the state estimation problem.

2.2 Depth estimation for aerial reconstruction

Depth estimation for aerial reconstruction has been an active research area for the past decade. This topic was pioneered by Mou et al. where a Convolutional Neural Network was used to construct height maps from single ortho images (Mou and Zhu, 2018).

The idea of using neural networks for aerial depth estimation has now been applied to many different neural architectures.

* Corresponding author
+ These authors contributed equally to this work

Currently, Neural Radiance Fields (Mildenhall et al., 2021) (NeRF) and Gaussian Splatting (Kerbl et al., 2023) architectures can be used to generate aerial ortho maps with very high fidelity. (Chen et al., 2024) use the NeRF architecture to render vertical view rays in a sparse voxel environment for efficient ortho map generation. (Yang et al., 2025) is a more recent architecture, that uses gaussian splatting instead. Both produce high-quality ortho images, but are not aimed at real-time use.

When it comes to real-time ortho image or DSM generation, current methods (Zhao et al., 2022) or (Kern et al., 2021) either use classical stereo-based matching methods (Geiger et al., 2011, Hirschmuller, 2008), or plane sweeping (Häne et al., 2014), respectively. Both have limitations: stereo-based matching methods require a specific geometric configuration between images, while plane sweeping, does not seem to generalise well to high-altitude scenes; see Figure 3.

2.3 Semantic segmentation for low-data remote sensing applications

Following the introduction of the UNet (Ronneberger et al., 2015) architecture, UNet-like models are introduced across many domains of application and heavily researched (Mo et al., 2022), as semantic segmentation datasets also start growing to meet the demand. Semantic segmentation sees its first foundational models in 2023, notably including the Segment Anything Model (Kirillov et al., 2023). Those models display cross-domain segmentation capabilities and can sometimes be queried for custom classes, but they typically do not apply well to remote sensing tasks due to the large difference between remote sensing images and everyday life pictures. Late 2023 (Jakubik et al., 2023) and 2024 (Wang et al., 2024) sees the rise of foundational models specifically trained for remote sensing, enabling more robust modeling for fields with less available data.

3. Platform

3.1 Hardware

The mapping platform is based on a fixed-wing airframe which is fitted with off-the-shelf hardware. ArduPilot is used as the flight control software and mission planner. A FLIR BlackFly U3-23-6C colour camera for monocular imaging and a Bosch BMI-088 IMU at 400 HZ for inertial measurements are used as sensors. The IMU is instrumented by a Teensy 4.0 ARM microcontroller. An NVIDIA Orin NX is used for onboard processing, which is an ARM-based System on Chip (SoC) with an integrated graphical processing unit (GPU). Since both the Central Processing Unit (CPU) and GPU are connected to the same physical memory, they can process images with 2048x1536 resolution at up to 20 frames per second(fps), depending on the configuration settings of the state estimation. The layout of the internal hardware can be seen in Figure 2, a full image of our UAV is show in Figure 1.

The camera was calibrated using TartanCalib (Duisterhof et al., 2022), which is an extension of the Kalibir (Rehder et al., 2016) calibration software. This extension uses an adaptive subpixel refinement window, which is more suitable for fisheye lenses. The IMU was calibrated using the ROS-Allan-Variance project (Buchanan, 2021). Finally, camera intrinsics refinement, Camera-IMU extrinsic estimation and Camera-IMU time offset were calibrated using Kalibr.

In addition to a UAV, a dedicated ground station is used for neural network inference and ortho mapping. The current specifications are an Intel i914900K CPU, a Nvidia RTX 5000 and 64 GB of RAM. Network communication is mainly facilitated via an Long Term Evolution (LTE) connection, however our system is also able to operate with a direct datalink antenna in areas where LTE is not available.

Figure 1. Fixed Wing UAV

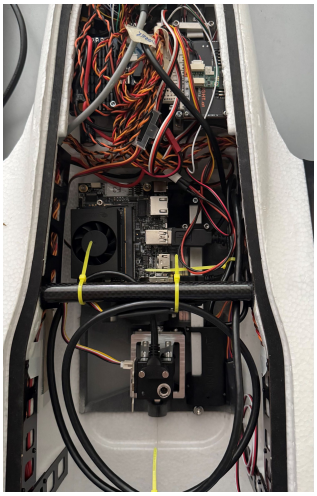


Figure 2. UAV - Internal. Bottom: Camera + IMU, Middle: Orin NX, Top: Flight Controller

3.2 Software

The software pipeline is composed of several modules running either on the UAV itself or the dedicated ground station. In order to facilitate communication between processes, either on the same machine, or on different physical devices connected by a network link, the Robot Operation System (ROS-Community, 2025) (ROS) is used, more specifically ROS2 Humble as that is the default version for the Orin NX. Special consideration is given to the IMU and Teensy. The Teensy microcontroller runs with micro-ROS (MicroROS-Community, 2025) which is ROS for embedded devices and is continuously synced to the Orin NXs clock.

The software pipeline is similar to (Fanta-Jende et al., 2023), with significant changes to the ROS module composition being described below. The initial sensor data i.e. GNSS, image

and IMU sensor data is passed to the pose estimation node running on the Orin NX. The pose estimation node utilises ORB-SLAM3 (OS3), which will be discussed in the next section 4.1 in more detail. Every image keyframe that has an associated trajectory value is sent to the georeferencing module for GNSS alignment to a specific coordinate frame (EPSG:3857) and then published over the network to the ground station. The keyframes are published at lower rate than the camera, which is at around 2 Hz to avoid congestion of the LTE data link.

On the ground station, the received georeferenced keyframes are given to the depth estimation module to compute a dense depth map of the image. For this step, the deep learning-based network CVA (Koestler et al., 2021) is integrated to produce fast and dense depth maps; which will be detailed in section 4.2. Following this, surface generation and ortho tile generation are performed similarly to prior work at a Ground Sampling Distance (GSD) if 0.15 meters per pixel. The ortho tiles are segmented and classified in real-time (Wang et al., 2024), as described in section 4.3. Finally, the ortho tiles are either embedded into a global image mosaic or tiled as 256x256 pixel subimages and sent to a Geographic Information Server (GIS), where the ortho image can be viewed via any software supporting the Web Map Tile Service (WMTS) (OGC, 2010).

4. Methodology

4.1 Monocular Visual-Inertial SLAM

ORB-SLAM3 (Campos et al., 2021), which was employed in previous works (Fanta-Jende et al., 2023), operated without the inclusion of an IMU. This framework includes an IMU and adapts the image processing to run at up to 20fps on an Nvidia Orin NX. This was achieved by extending the work of (Longyong, 2021) which implements ORB feature descriptors as Compute Unified Device Architecture (CUDA) kernels. However, memory bandwidth remains a bottleneck in these applications as by default the data needs to be explicitly copied to be accessible on the GPU. To solve this problem, the kernels are adapted to make use of CUDA unified memory. This allows an application to have the same memory address be accessible by the CPU and GPU. On memory access, the driver may page-fault and trigger a copy to the desired device. However, since the Orin NX devices have a shared physical memory pool (Section 3.1), CUDA kernels without any additional memory copies can be used for higher performance.

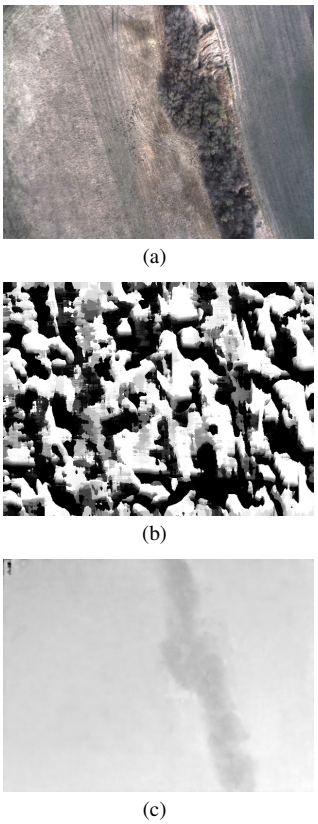
In Section 5, our results are compared to OpenVSLAM (OVS) and OS3 as those served as the baseline state estimation algorithms in prior works. Both SLAM trajectories are aligned with the GNSS measurements using a sliding window Umeyama alignment of 60 measurements. More specifically, image and GNSS measurements with the smallest time delta are paired and collected until 60 pairs are recorded. These measurements are given to the Umeyama alignment algorithm for alignment estimation. Once a newer image/GNSS pair arrives, the oldest one is discarded. Loop closure is disabled due to the scale change it would produce for established trajectory segments, which would result in an re-evaluation of the entire map generation (Liu et al., 2025).

4.2 Depth Estimation Network

CVA (Koestler et al., 2021) is integrated for dense depth reconstruction in real-time. This approached is compared to a GPU-accelerated plane sweep algorithm (Häne et al., 2014), which slices the space into discrete planes along the viewing-ray of the camera and computes a pixel warping using a photometric correlation function.

The CVA model (Koestler et al., 2021) is based on the plane sweeping algorithm with the plane distance as a learnable parameter. Images and camera poses are provided as inputs to the model. The network supports odd numbers of image sets (i.e. 3, 5, 7...) where the center image is considered the reference view. Giving the camera poses as a prior enables a more flexible geometric configuration of the supplied trajectory for which images do not have to be rectified i.e. warped such that the epipolar lines between image pairs are horizontal and parallel. (Zhao et al., 2022). Furthermore, as this method directly yields dense depth maps from the network, a sparse point cloud generated by a structure-from-motion or SLAM pipeline (Zhao et al., 2022, Liu et al., 2025) is not required. For this pipeline, three consecutive images are sufficient for dense depth maps to be reconstructed in metric scale at real-time speeds (i.e below 100ms per image), from the georeferenced poses of overlapping keyframe images.

A qualitative example can be seen in Figure 3. Depth estimation with plane sweeping seems to fail when processing large planar area of homogeneous texture (such as fields or meadows), while the learned network accurately reconstructs the depth of the observed image. To our knowledge, our work is the first that uses an estimation network for depth map generation in a real-time ortho-rectification pipeline.



(a) (b) (c)

Figure 3. (a) Camera Image (b) Depth Image with Plane Sweeping (c) Depth Image with the CVA Model

4.3 Scene Understanding

This work aims to provide first responders with semantic understanding of the terrain in real-time. For this purpose, semantic

segmentation is achieved for basic landcover classes (trees, low vegetation, roads, and buildings) and two scenario-relevant classes (debris and destroyed buildings). These present a significant challenge as training and validation data are scarce. Aerial datasets for basic landcover classes are to some extent comprehensive, including for instance the Semantic Drone Dataset (Semantic Drone Dataset, 2025) (not used in this study due to very high and somewhat inconsistent GSD), LandCoverAI (Boguszewski et al., 2021), ISPRS Potsdam (Rottensteiner et al., 2014a) and ISPRS Vaihingen (Rottensteiner et al., 2014b). However, annotated data for scenario-relevant classes mostly comprise FloodNet (Rahnemoonfar et al., 2021) and RescueNet (Rahnemoonfar et al., 2023), two datasets which both present similar issues - they both feature a single acquisition location, leading to very similar images and high contamination between different splits. Their annotations are rather coarse and contain errors, and overall they represent but a tiny proportion of the possible variation factors for disaster areas.

To train the model despite the lack of data and the expected variability in the input space, both a pre-trained state-of-the-art foundation model from (Wang et al., 2024), and a concatenation of various datasets are leveraged. The data is aligned from these different datasets by resampling them to similar GSDs, and the frequency of each dataset during training is controlled.

The limited availability of scenario-specific data is addressed by generating synthetic scenes using Blender (Community, 2018) for procedural rendering and the Stable Diffusion 2 Depth model, based on Latent Stable Diffusion (Rombach et al., 2022), for image refinement and relighting. This pipeline enables fast generation of realistic top-down imagery with accurate labels under controlled conditions. Procedural scenes in Blender are first constructed using geometry nodes. Each scene includes terrain with varying textures (e.g., grass, road), vegetation (trees), and buildings, which can appear either intact or destroyed. Scene diversity is introduced via Perlin noise-based segmentation masks that control the spatial layout of textures and object placement. Texture colours, object locations, and terrain elevation (e.g., flat vs. hilly) are randomised to simulate varied environments. The camera remains fixed with a nadir view, while lighting angles are varied to simulate different times of day. To simulate destruction, random circular region is defined in each scene where debris and rubble are scattered. After rendering, the RGB images are enhanced using the depth-aware Stable Diffusion 2 Depth model. The model estimates monocular depth and uses it to guide relighting and realism improvements based on text prompts. For each scene, both an intact and a destroyed version are generated using the following prompts: "*Satellite image of houses, roads, trees and fields.*" and "*Satellite image of damaged houses and roads*".

Finally, the predefined destruction area is used to blend the intact and damaged versions into a realistic composite depicting localised structural damage. Figure 4 illustrates the pipeline output, from the initial render to the final composite image. The final enhanced render can then be paired with the corresponding labeled mask for downstream tasks.

5. Experiments and Results

Current UAV datasets lack IMU integration (Rahnemoonfar et al., 2023, Montgomery et al., 2021, Bu et al., 2016) which makes it challenging for us to evaluate our reconstruction pipeline. For this reason, a specific dataset is acquired and used as a benchmark for our reconstruction pipeline. The flights are conducted with the fixed-wing UAV described in Section 3.1 (see also Figure 1) at 90m height with a speed of approximately 20 m/s, Recording a UAV flight yields a reference trajectory (Figure 8), ortho image and DSM. Furthermore, we surveyed ground control points (GCPs), as well as post-processed the recorded data using Pix4D. A post-processed image mosaic is shown in Figure 5, while the output of our real-time pipeline is shown in Figure 6.

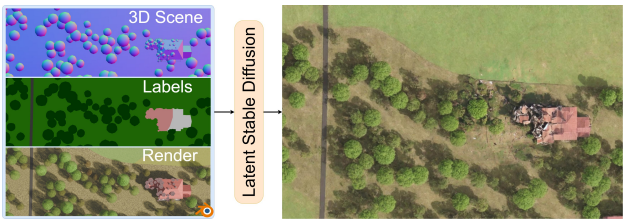


Figure 4. Image generation pipeline using Blender and Latent Stable Diffusion

The state estimation running on the UAV varies between 10-20fps depending on the given settings. For this evaluation, the recorded data was processed at 50% image resolution, 10 fps, 2 pyramid levels, with a scale factor of 2 and 3000 features per level. Even though we have a fixed flight altitude we use image pyramid levels to compensate for our camera motion blur (Klein and Murray, 2008).

An overlay of the realtime ortho and the Pix4D ortho, optimised with GCPs, can be seen in Figure 7. As each image from the UAV is processed on the fly and only aligned locally to the Umeyama window, along with using raw GNSS measurements, local inconsistencies in the realtime ortho arise.

Since we are targeting a timing value of 2Hz i.e. 2 frames per second of data output from our UAV, we have 500ms to process each image on the ground station. Any processing times higher than that will cause an accumulation of data in the internal messaging queues and over time and throttle the map output.

Timings for processing the individual stages of our pipeline are given in Table 1. We can report sub 500ms timings for all our processing stages. We have a fixed offset of around 2 seconds for the densification stage, since it requires 3 images which have to be accumulated. As a result, as soon as our UAV captures an image, the tile information associated with that image is published seconds later.

Stage	Time (ms)
(UAV) State Estimation & Georeferencing	$\approx$ 70-200
(Ground) Densification	$\approx$ 100
(Ground) Surface Generation	$\approx$ 440
(Ground) Ortho-rectification + Segmentation	$\approx$ 65
(Ground) Mosaicing	$\approx$ 70
Total processing time per frame	$\approx$ 945-875

Table 1. Table showing the end-to-end processing time for each stage in the pipeline. End-to-end is defined as the time from data reception by the ROS node until data publishing. Network transmission time is excluded

Table 2 shows the Root Mean Square Error (RSME) between the GNSS and GNSS-aligned SLAM trajectories, where an improvement of the RMSE up to x60 using an IMU can be seen.

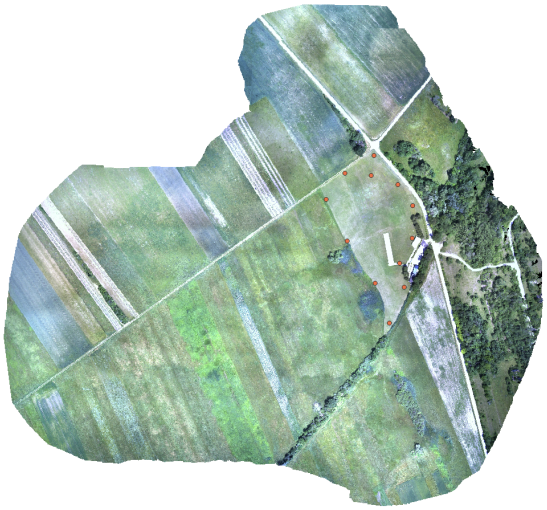


Figure 5. Ortho mosaic reconstruction using Pix4d with GCP optimisation. GCPs are marked as pink dots

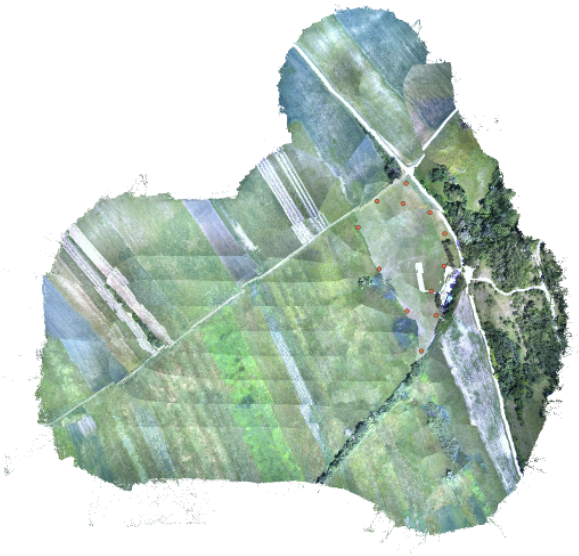


Figure 6. Ortho mosaic reconstruction using our realtime pipeline.

Table 3 compares the resulting XY error of the measured GCPs. The average error is around 5 meters in XY which corresponds to the error of raw GNSS measurements as RTK corrected data is not used. When using RTK, higher absolute (and relative) accuracies can be achieved (Kern et al., 2021, Fanta-Jende et al., 2023) - this may also be reflected in a greater consistency of the data products. The XY errors for a Pix4d ortho image which was processed on the same raw GNSS data, without GCP adjustment are given in Table 4. Our results are marginally worse, but computed from a realtime system.

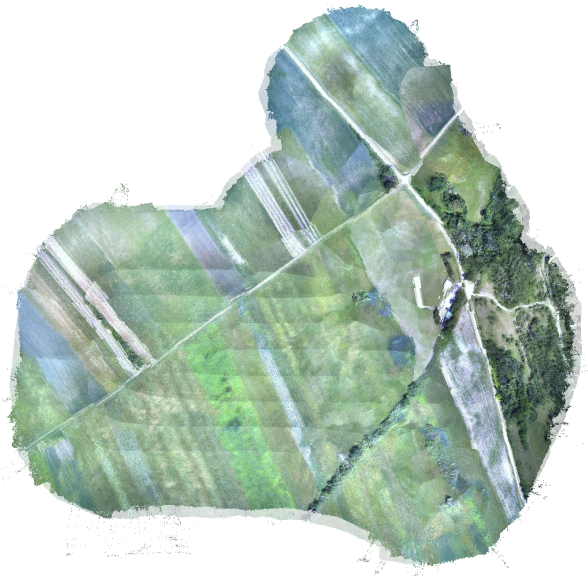


Figure 7. Pix4D GCP optimised ortho with 30% transparency overlaid over the realtime ortho

RSME (m)	OVS	OS3	OS3 With IMU
X	15.117	13.541	0.242
Y	10.742	8.823	0.197
Z	0.763	0.535	0.161
XYZ	18.560	16.171	0.351

Table 2. Comparing the Root Mean Square Error Between GNSS and GNSS Aligned SLAM Trajectories

Point	△ XY (m)	Point	△ XY (m)
GCP1	4.5	GCP 7	0.73
GCP2	7.43	GCP 8	2.93
GCP3	(GCP not visible)	GCP 9	6.34
GCP4	4.06	GCP 10	5.94
GCP5	8.31	GCP 11	4.73
GCP6	5.8	GCP 12	2.3
Avg RSME XY	4.82		

Table 3. Showing the XY error between measured GCPs and corresponding marker in realtime ortho

Point	△ XY (m)	Point	△ XY (m)
GCP1	3.17	GCP 7	3.04
GCP2	3.25	GCP 8	3.02
GCP3	3.18	GCP 9	3.04
GCP4	3.2	GCP 10	3.13
GCP5	3.12	GCP 11	3.16
GCP6	3.06	GCP 12	3.18
Avg RSME XY	3.13		

Table 4. Showing the XY error between measured GCPs and corresponding marker in Pix4D ortho without GCP optimisaiton

Our landcover segmentation model is trained using 5 datasets as described in 4.3. The strength and weaknesses of these datasets are adjusted by over-sampling the datasets with respect to

Figure 8. Flight trajectory over ORB-SLAM3 data output. The pointcloud is not used for 3D reconstruction in our pipeline.

LandCoverAI (Boguszewski et al., 2021). The ISPRS Potsdam (Rottensteiner et al., 2014a) and ISPRS Vaihingen (Rottensteiner et al., 2014b) datasets, presenting some rare urban environments to train on, are seen with a probability 16 times higher during training. Being our only available datasets with scenario-specific classes, our synthetic dataset and RescueNet (Rahnemoonfar et al., 2023) are respectively over-sampled by a factor of 4 and 8.

Landcover segmentation suffers from the same limitations for validation as for training, namely the difficulty to acquire appropriate data for disaster-related classes. When it comes to standard landcover classes, a new dataset is acquired to confirm the model's robustness to different sensors, environments and lighting conditions. For destroyed buildings and debris, the model is evaluated on the validation sets from our synthetic set and RescueNet, while acknowledging that subsequent proof is needed to conclude on the capacity of the model to recognise infrastructure damage in general. In Figure 9, quantitative results are shown on each class for each dataset. The results are overall very good for the validation splits of datasets present at training, and significantly dip for roads and buildings when generalising to our more challenging dataset. Still, considering the challenges of artifacts and high exposure (see Figure 10 for visual examples), the reported performance is much higher than previous models on this dataset (Fanta-Jende et al., 2023). In our opinion, this study brings to light a gap in data availability when it comes to segmenting infrastructure damage from aerial images. It also questions the diversity of datasets such as RescueNet, where images in the training and validation splits remain very similar.

Figure 11 shows the output of the semantic mapping, computed in real-time along with the other outputs of our pipeline. The main areas of vegetation and roads were identified. The major source of error comes from the miss-classification of high-reflectance areas and path-like structures in crop fields as road

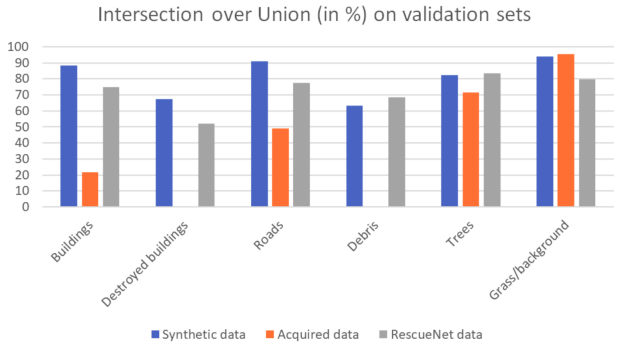


Figure 9. Quantitative results on all our validation sets.

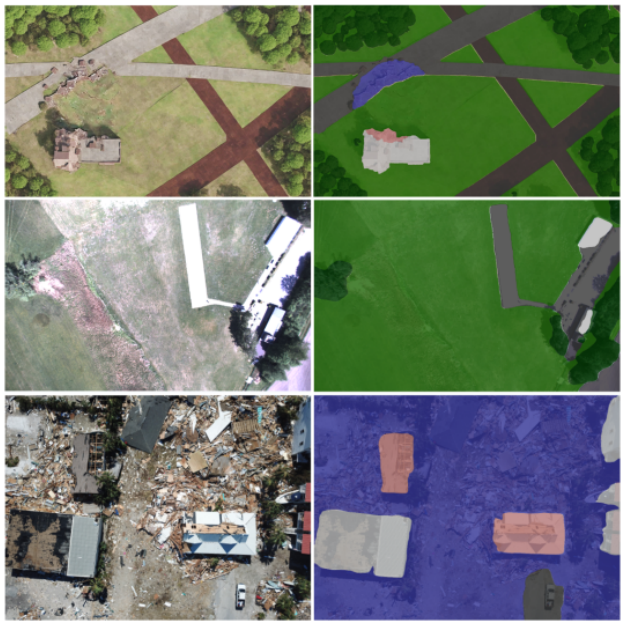


Figure 10. Left: sample images from the validation splits of our synthetic dataset, our acquired dataset and RescueNet (top to bottom). Right: overlays with predicted class (Buildings: light grey; Destroyed buildings: pink; Roads: dark grey; Debris: blue; Trees: dark green; Grass: light green).

instead of low vegetation. Furthermore, the building adjacent to the landing strip was only partially classified, and several smaller patches were incorrectly labeled as water. These errors can probably be ascribed to the visual artifacts due to the raw GNSS measurements. Considering these artifacts and the significant differences between this data and the training sets, the model seems to significantly out-perform previous attempts in terms of its capability to generalise. A visual fit of our segmentation against our real-time orthorectified map can be seen in Figure 12.

6. Conclusion

This work illustrates the feasibility of an end-to-end real-time ortho and scene understanding pipeline. We report errors in a range that is acceptable for raw GNSS measurements, as well as shed light on the performance of Foundation Models for semantic segmentation when existing datasets have gaps. In order to reduce the current errors and improve the global consistency of the generated map, this pipeline will be tested with higher accuracy GNSS measurements as well as with a global optim-

Figure 11. Ortho mosaic with realtime segmentation. **Green** - Grass, **Dark Green** - Trees, **Grey** - Road, **Blue** - Water, **Red** - Building.

Figure 12. Segmentation image overlayed with the realtime ortho map with an opacity of 30%.

isation step. Future works on the semantic pipeline should involve a more comprehensive assessment and data acquisition effort for classes relevant to crisis and disaster scenarios such as destroyed or flooded infrastructure.

7. Acknowledgements

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

References

Boguszewski, A., Batorski, D., Ziemba-Jankowska, N., Dziedzic, T., Zambrzycka, A., 2021. Landcover. ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1102–1110.

Botterill, T., Mills, S., Green, R., 2010. Real-time aerial image mosaicing. *2010 25th International Conference of Image and Vision Computing New Zealand*, 1–8.

Bu, S., Zhao, Y., Wan, G., Liu, Z., 2016. Map2dfusion: Real-time incremental uav image mosaicing based on monocular slam. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4564–4571.

Buchanan, R., 2021. Allan variance ros. Oxford Robotics Institute, DRS Lab.

Campos, C., Elvira, R., Rodríguez, J. J. G., M. Montiel, J. M., D. Tardós, J., 2021. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Transactions on Robotics*, 37(6), 1874-1890.

Chen, S., Yan, Q., Qu, Y., Gao, W., Yang, J., Deng, F., 2024. Ortho-NeRF: generating a true digital orthophoto map using the neural radiance field from unmanned aerial vehicle images. *Geo-spatial Information Science*, 1-20.

Community, B. O., 2018. Blender - a 3D modelling and rendering package. Blender Foundation, Stichting Blender Foundation, Amsterdam.

Duisterhof, B. P., Hu, Y., Teng, S. H., Kaess, M., Scherer, S., 2022. Tartancalib: Iterative wide-angle lens calibration using adaptive subpixel refinement of apriltags.

Erdelj, M., Natalizio, E., Chowdhury, K. R., Akyildiz, I. F., 2017. Help from the Sky: Leveraging UAVs for Disaster Management. *IEEE Pervasive Computing*, 16(1), 24-32.

Fanta-Jende, P., Steininger, D., Kern, A., Widhalm, V., Apud Baca, J. G., Hofstätter, M., Simon, J., Bruckmüller, F., Sulzbachner, C., 2023. Semantic Real-Time Mapping with UAVs. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 91(3), 157–170. https://doi.org/10.1007/s41064-023-00242-2.

Geiger, A., Roser, M., Urtasun, R., 2011. Efficient large-scale stereo matching. R. Kimmel, R. Klette, A. Sugimoto (eds), *Computer Vision – ACCV 2010*, Springer Berlin Heidelberg, Berlin, Heidelberg, 25–38.

Hermann, M., Weinmann, M., Nex, F., Stathopoulou, E., Remondino, F., Jutzi, B., Ruf, B., 2024. Depth estimation and 3D reconstruction from UAV-borne imagery: Evaluation on the UseGeo dataset. *ISPRS Open Journal of Photogrammetry and Remote Sensing*, 13, 100065.

Hirschmuller, H., 2008. Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2), 328-341.

Häne, C., Heng, L., Lee, G. H., Sizov, A., Pollefeys, M., 2014. Real-time direct dense matching on fisheye images using plane-sweeping stereo. *2014 2nd International Conference on 3D Vision*, 1, 57–64.

Jakubik, J., Chu, L., Fraccaro, P., Bangalore, R., Lambhate, D., Das, K., Oliveira Borges, D., Kimura, D., Simumba, N., Szwarcman, D., Muszynski, M., Weldemariam, K., Zadrozny, B., Ganti, R., Costa, C., Watson, C., Mukkavilli, K., Roy, S.,

Phillips, C., Ankur, K., Ramasubramanian, M., Gurung, I., Leong, W. J., Avery, R., Ramachandran, R., Maskey, M., Olofossen, P., Fancher, E., Lee, T., Murphy, K., Duffy, D., Little, M., Alemohammad, H., Cecil, M., Li, S., Khallaghi, S., Godwin, D., Ahmadi, M., Kordi, F., Saux, B., Pastick, N., Doucette, P., Fleckenstein, R., Luanga, D., Corvin, A., Granger, E., 2023. HLS Foundation.

Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4). https://reposam.inria.fr/fungraph/3d-gaussian-splatting/.

Kern, A., Bobbe, M., Khedar, Y., Bestmann, U., 2020. Openrealm: Real-time mapping for unmanned aerial vehicles. *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 902–911.

Kern, A., Fanta-Jende, P., Glira, P., Bruckmüller, F., Sulzbachner, C., 2021. AN ACCURATE REAL-TIME UAV MAPPING SOLUTION FOR THE GENERATION OF ORTHOMOSAICS AND SURFACE MODELS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2021, 165–171. https://isprs-archives.copernicus.org/articles/XLIII-B1-2021/165/2021/.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y. et al., 2023. Segment anything. *Proceedings of the IEEE/CVF international conference on computer vision*, 4015–4026.

Klein, G., Murray, D., 2008. Improving the agility of keyframe-based slam. D. Forsyth, P. Torr, A. Zisserman (eds), *Computer Vision – ECCV 2008*, Springer Berlin Heidelberg, Berlin, Heidelberg, 802–815.

Koestler, L., Yang, N., Zeller, N., Cremers, D., 2021. Tandem: Tracking and dense mapping in real-time using deep multi-view stereo. *Conference on Robot Learning (CoRL)*.

Liu, Y., Akbar, A., Yu, T., Yu, Y., Kong, Y., Gao, J., Wang, H., Li, Y., Zhao, H., Liu, C., 2025. Artemis: A real-time efficient ortho-mapping and thematic identification system for uav-based rapid response. Unpublished manuscript.

Longyong, W., 2021. Orb slam 3 mokai. https://github.com/SYSU-FishTouchers/ORB_SLAM3_MOKAI. (04-07-2025).

MicroROS-Community, 2025. micro.ros.org. https://micro.ros.org/. (17-06-2025).

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., Ng, R., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1), 99–106.

Mo, Y., Wu, Y., Yang, X., Liu, F., Liao, Y., 2022. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing*, 493, 626–646.

Montgomery, J., Wartman, J., Reed, A. N., Gallant, A. P., Hutabarat, D., Mason, H. B., 2021. Field reconnaissance data from GEER investigation of the 2018 MW 7.5 Palu-Donggala earthquake. *Data in Brief*, 34, 106742.

Mou, L., Zhu, X., 2018. Im2height: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network.

OGC, 2010. OGC Web Map Tile Service (WMTS) Standard. OGC Standard. Version 1.0.0, OGC 07-057r7.

Rahnemoonfar, M., Chowdhury, T., Murphy, R., 2023. RescueNet: A High Resolution UAV Semantic Segmentation Dataset for Natural Disaster Damage Assessment. *Scientific Data*, 10(1), 913. https://doi.org/10.1038/s41597-023-02799-4.

Rahnemoonfar, M., Chowdhury, T., Sarkar, A., Varshney, D., Yari, M., Murphy, R. R., 2021. Floodnet: A high resolution aerial imagery dataset for post flood scene understanding. *IEEE Access*, 9, 89644–89654.

Rehder, J., Nikolic, J., Schneider, T., Hinzmann, T., Siegwart, R., 2016. Extending kalibr: Calibrating the extrinsics of multiple imus and of individual axes. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 4304–4311.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B., 2022. High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, Springer, 234–241.

ROS-Community, 2025. Ros 2 documentation: Humble documentation. https://docs.ros.org/en/humble/index.html. (16-06-2025).

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., 2014a. Isprs semantic labeling contest. *Proceedings of the ISPRS Commission III Symposium*, 1, ISPRS, Leopoldshöhe, Germany.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., 2014b. Isprs semantic labeling contest – vaihingen dataset. *Proceedings of the ISPRS Commission III Symposium*, 1, ISPRS, Leopoldshöhe, Germany.

Semantic Drone Dataset, 2025. http://dronedataset.icg.tugraz.at. (2025-06-25).

Wang, D., Zhang, J., Xu, M., Liu, L., Wang, D., Gao, E., Han, C., Guo, H., Du, B., Tao, D., Zhang, L., 2024. MTP: Advancing Remote Sensing Foundation Model Via Multi-Task Pretraining. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 1-24.

Wang, W., Zhao, Y., Han, P., Zhao, P., Bu, S., 2019. Terrainfusion: Real-time digital surface model reconstruction based on monocular slam. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 7895–7902.

Yang, J., Cai, Z., Wang, T., Ye, T., Gao, H., Huang, H., 2025. Ortho-3DGS: True Digital Orthophoto Generation From Unmanned Aerial Vehicle Imagery Using the Depth-Regulated 3D Gaussian Splatting. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 18, 10972-10994.

Zhao, Y., Chen, L., Zhang, X., Xu, S., Bu, S., Jiang, H., Han, P., Li, K., Wan, G., 2022. RTSfM: Real-Time Structure From Motion for Mosaicing and DSM Mapping of Sequential Aerial Images With Low Overlap. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 1-15.