sciendo

Reports on Geodesy and Geoinformatics, 2022, Vol. 114, pp. 15–23

DOI: 10.2478/rgg-2022-0006
Received: 8 July 2022 / Accepted: 1 December 2022
Published online: 20 December 2022

ORIGINAL ARTICLE

Vehicle detection and masking in UAV images using YOLO to improve photogrammetric products

Karolina Pargieła [1*]

[1]Department of Photogrammetry, Remote Sensing of Environment, and Spatial Engineering, Faculty of Geo-Data Science, Geodesy, and Environmental Engineering, AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Krakow, Poland

[*]pargiela@agh.edu.pl

Abstract

Photogrammetric products obtained by processing data acquired with Unmanned Aerial Vehicles (UAVs) are used in many fields. Various structures are analysed, including roads. Many roads located in cities are characterised by heavy traffic. This makes it impossible to avoid the presence of cars in aerial photographs. However, they are not an integral part of the landscape, so their presence in the generated photogrammetric products is unnecessary. The occurrence of cars in the images may also lead to errors such as irregularities in digital elevation models (DEMs) in roadway areas and the blurring effect on orthophotomaps. The research aimed to improve the quality of photogrammetric products obtained with the Structure from Motion algorithm. To fulfil this objective, the Yolo v3 algorithm was used to automatically detect cars in the images. Neural network learning was performed using data from a different flight to ensure that the obtained detector could also be used in independent projects. The photogrammetric process was then carried out in two scenarios: with and without masks. The obtained results show that the automatic masking of cars in images is fast and allows for a significant increase in the quality of photogrammetric products such as DEMs and orthophotomaps.

Key words: deep learning, photogrammetry, Structure from motion, roads, object detection

1 Introduction

Unmanned Aerial Vehicles (UAVs), due to their affordability and the ability to generate accurate photogrammetric products, are used in many areas, including land surveying (Koeva et al., 2018; Šafář et al., 2021), agriculture (Delavarpour et al., 2021; Kaivosoja et al., 2021), forestry (Dainelli et al., 2021; Pessacg et al., 2022), mining (Ge et al., 2016; Park and Choi, 2020), archeology (Campana, 2017; Fiz et al., 2022), construction (Li and Liu, 2019; Yahya et al., 2021), and road construction (Cardenal et al., 2019; Zulkipli and Tahar, 2018). Furthermore, 3D modelling of the existing infrastructure is crucial for developing increasingly popular smart cities. The use of UAVs allows for rapid and relatively inexpensive acquisition of a significant amount of data from the study area, compared to other measurement methods (Roberts et al., 2020).

The Structure from Motion – MultiView Stereo (SfM-MVS) technique is mostly used to process UAV data (Carrera-Hernández et al., 2020; Eltner and Sofia, 2020; Nyimbili et al., 2016; Snavely et al., 2008). SfM allows for automatic orientation of images while MVS enables 3D reconstruction of a captured scene. SfM features several properties that are essential for handling large sets of UAV images. These are: high level of automation, ability to handle perspective and scale changes, processing data without prior knowledge about neither interior camera parameters, nor exterior image orientation. To ensure successful performance of this method, the images used for reconstruction should be characterised by sufficiently high overlap (at least 60-70%, typically 80-90%). At first, SfM searches for features in the whole image collection and describes them, for example using scale-invariant feature transform (SIFT) detector and descriptor. Then, descriptors are matched, and image point to image point correspondences are established. SfM is an incremental algorithm which forms an image block by successively orienting new images, typically using 2D-3D correspondences (resection), along with triangulating new object points (forward intersection).

Random sample consensus (RANSAC) or other sample consensus based algorithms are applied to eliminate outlier matches. This process is interleaved with robust Bundle Adjustment (BA) to improve accuracy in intermediate steps. As a result, SfM provides fairly accurate estimations of camera positions and rotation angles, along with a collection of object points (Structure), that is sometimes referred to as a sparse point cloud. The alignment of an image block can be further enhanced by running BA with ground control points (GCPs), and image positions and orientation angles provided by GNSS/INS system (if available). Once the camera positions and rotation angles have been accurately determined, it is possible to proceed to dense feature matching using MVS techniques to generate a dense point cloud (Bianco et al., 2018; Jiang et al., 2020; Iglhaut et al., 2019).

The rapid development of infrastructure and the need to ensure safety of road systems require frequent measurements related to monitoring, design, construction, and maintenance. Topographic information is typically obtained through land surveying devices, i.e. total stations or GNSS receivers. They allow for reaching high measurement accuracy but also require significant human and time resources. Furthermore, terrestrial measurements highly depend on the human factor which may result in systematic errors. It can also pose problems to directly access some areas in the terrain of varied characteristics. The use of UAVs allows for quick data acquisition over vast areas. At the same time, it is much safer than terrestrial measurements when working on roads or areas with varied relief (Cardenal et al., 2019; Tan and Li, 2019; Zulkipli and Tahar, 2018).

Masking of cars and other land vehicles in images can be performed manually. However, this is a lengthy process, especially when processing highly congested roads. In both theory and practice, tendencies towards enhancing process automation can be noticed. There is a desire for replacing human work with the use of algorithms and for reducing the working time of operators (Coombs et al., 2020; Gruen, 2021). Vehicles can be detected in images much more efficiently by implementing automatic detection and subsequent processing of the results into masks (Yang et al., 2021; Zhu et al., 2021). Currently, numerous algorithms are available for object detection in images. Some of them are based on a multi-step operation involving information area selection, feature extraction, and classification (Xiao et al., 2020; Zhao et al., 2019). With the development of deep neural networks, Convolutional Neural Networks (CNN) algorithms have gained popularity (Indolia et al., 2018; Han et al., 2021; Li and Lin, 2020). Additionally, to address the problem of identifying many objects in one image, the Region-based Convolutional Neural Networks (R-CNN) solution was introduced. It is also possible to use a CNN convolutional neural network combined with the region proposal algorithm which poses the object location hypothesis (Girshick et al., 2014). This technique was developed into an improved algorithm called Fast R-CNN (Girshick et al., 2015). Its implementation aimed at mitigating errors of previous algorithms by building a faster algorithm based on a convolutional feature map generated directly from an input image. Another solution was the Faster R-CNN approach (Ren et al., 2017), in which an image is also provided as input to the convolutional network. However, it is not based on a selective search like the two previous versions, but on the introduction of Regional Proposal Network (RPN). One of the most commonly used object detection algorithms is YOLO (You Only Look Once). It is a state-of-the-art approach that allows for a one-step operation, in which a region proposal field is generated during classification. This means that a single convolutional network simultaneously predicts multiple bounding boxes and class probabilities. Thanks to the detection speed, the algorithm is also used for real-time operations (Gromada et al., 2022; Koay et al., 2021; Redmon et al., 2016). The algorithm and its modifications are widely used for object detection in images acquired with UAVs (Bao et al., 2021; Luo et al., 2020; Sahin and Ozer, 2021; Tan et al., 2021).

Aerial data acquisition using UAVs is connected with capturing cars present in a study area. They are not integral landscape elements thus their presence in photogrammetric products is unnecessary. Additionally, in the case of digital elevation models, they may generate errors impossible to detect without original images showing exact car locations. This paper presents an approach using the YOLO v3 object detection algorithm to mask cars in images acquired with UAVs. It also describes the photogrammetric processing that was performed on the images with and without masks, and the digital elevation model and orthophotomap generation. The research aimed to show that automatic masking of cars on UAV images allows for improving the quality of such products as digital elevation models and orthophotomaps. An additional research aspect was the optimisation problem. Manual car masking is a highly time-consuming process, particularly when mapping roads with heavy traffic. The use of the YOLO detector makes it possible to create a detection model that can be used in other projects without the need to duplicate the activities associated with manual masking of cars.

Figure 1. Used DJI Phantom 4Pro v2.0 UAV

2 Materials and Methods

2.1 Fieldwork

A photogrammetric flight for the study was carried out with a DJI Phantom 4 Pro v2.0 (Figure 1). This device is equipped with a 1-inch, 20-megapixel CMOS global shutter camera with gimbal stabilisation. The camera is equipped with an f = 8.8 mm lens, which corresponds to f = 24 mm lens for a full frame format. The images were stored in the JPEG files, with full resolution of 5472 × 3648 pixels. Exposure parameters were set to automatic. The UAV was not equipped with an RTK system, so the coordinates of the projection centres acquired with the navigation module were not taken into account in the alignment process. The 4S LiPo battery supplied power allowing for up to 30 minutes of an uninterrupted flight.

The photogrammetric data acquisition was carried out on 21st August 2021, in the city of Kraków (50°03′41″N 19°56′18″E). The measurement structure was a 700 m long and 50 m wide road (Figure 2). The photogrammetric flight included additional strips of approximately 40 m next to the road. Due to its geometry, with one dimension significantly exceeding the other, the measurement structure can be classified as linear. In order to enable subsequent georeferencing and accuracy checks, 32 ground control

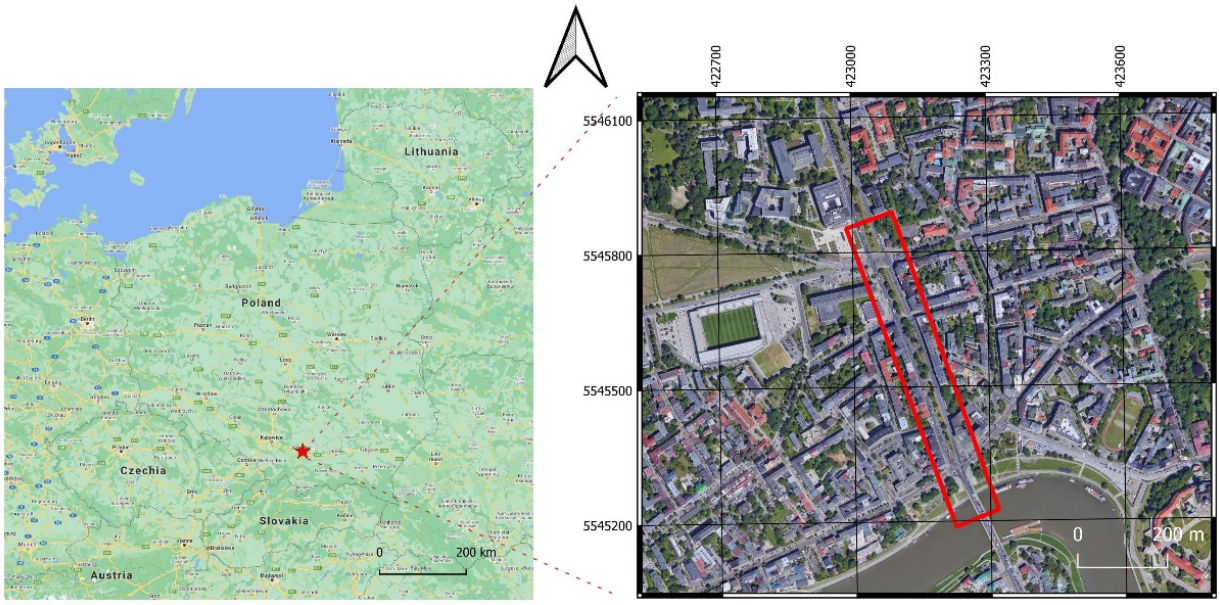

Figure 2. Study area. Coordinates are referred to UTM Zone 34 N (EPSG: 32634). Background image: Google Earth, earth.google.com/web/.

points (GCPs) were measured with attempting to maintain their regular distribution throughout the study area. Only natural points were used. The survey was performed using the RTK technique with the use of two Leica GS16 receivers, one as a rover and the other one as a base in the immediate vicinity of the research area. The coordinates of the base were adjusted based on the coordinates of the ASG-EUPOS permanent system station KRA1, located 765 meters from the base. The coordinates of GCPs were refined based on corrections received from the adjustment of the KRA1 – base vector. Planimetric (X, Y) coordinates of the GCPs were defined in the ETRS89 / Poland CS2000 zone 7 (EPSG: 2178) coordinate system, and their height (Z) was determined in the Kronstadt 86 height/vertical reference system. The research area is located in the city centre and covers a highly congested road, which was a key factor for the selection of this area as a test field for the conducted research.

The photogrammetric flight was designed and performed using the DJI Pilot application. The endlap and sidelap were set to 80%. The flight was made in four rows, nadiral images were taken at a height of 68 m above ground, which results into a ground sample distance of 1.86 cm/pixel. Due to the limited battery life and the significant distance of the flight in the city centre, it was divided into two parts to avoid loss of visibility and communication with the aircraft. A total of 216 images were obtained.

2.2 YOLO v3 algorithm

The YOLO v3 algorithm (Redmon and Farhadi, 2018) was used to detect cars in the images. The algorithm is based on the CNN and allows for the detection of objects both in images and in real-time in videos. The algorithm enables the detection of multiple objects in single images by assigning them specific classes and indicating their locations in the bounding box. The evaluation of regions is performed on the basis of their similarity to predefined classes. The algorithm works by dividing the image into a grid in which the region's probability is predicted for each cell. Next, the fitting of bounding boxes is performed, whose size is determined by 4 coordinates: tx, ty, tw, th. Object confidence and class predictions are calculated using logistic regression, which allows for the correct prediction of objects belonging to more than one class. The selection of a single frame from the set detected for a given object is based on filtering frames with a too low level of objectivity, and then the Non-maximum Suppression method is used (Horzyk and Ergün, 2020). YOLO v3 uses Darknet-53 containing 53 convolutional layers for feature extraction. The algorithm uses consecutive 3x3 and 1x1 convolutional layers, and the idea of residual networks to increase the depth of the network and prevent a vanishing gradient. YOLO v3 trains the network by decreasing the loss function values by evaluating the actual and predicted model values based on Equation (1) (Wang et al., 2022):

$$
\begin{aligned}
Loss = {} & \lambda coord \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\left(x_i^j - \hat{x}_i^j \right)^2 + \left(y_i^j - \hat{y}_i^j \right)^2 \right] \\
& + \lambda coord \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\left(\sqrt{w_i^j} - \sqrt{\hat{w}_i^j} \right)^2 + \left(\sqrt{h_i^j} - \sqrt{\hat{h}_i^j} \right)^2 \right] \\
& - \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\hat{C}_i^j \log \left(C_i^j \right) + \left(1 - \hat{C}_i^j \right) \log \left(1 - C_i^j \right) \right] \\
& - \lambda noobj \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} \left[\hat{C}_i^j \log \left(C_i^j \right) + \left(1 - \hat{C}_i^j \right) \log \left(1 - C_i^j \right) \right] \\
& - \sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{C \in classes} \left(\left(\hat{P}_i^j \log \left(P_i^j \right) + \left(1 - \hat{P}_i^j \right) \log \left(1 - P_i^j \right) \right) \right)
\end{aligned}
\tag{1}
$$

where: x_i^j, y_i^j represent coordinates of the centre of the bounding box, w_i^j, h_i^j represent the size of the bounding box (width and height), S represents the grid size, B represents the number of prediction frames, I_{ij}^{obj} and I_{ij}^{noobj} represent probability that box appears or has no target at i,j, C_i^j, $\hat{C}_i^j$ represent box confidence score in a cell and box confidence score for the predicted object, P_i^j, $\hat{P}_i^j$ represent the predicted and actual value of target probability.

In addition, the prediction is based on three different scales to detect objects of different sizes, taking steps 32, 16, and 8 respectively. For example, the input image size of 416x416 will result in grids of 13x13, 26x26, and 52x52 respectively. The fusion of functions is based on upsampling so that feature maps from all scales have the same size. Combining features from previous layers is performed by concatenation (Ju et al., 2019; Xu et al., 2020). The workflow of the YOLO v3 algorithm is shown in (Figure 3).

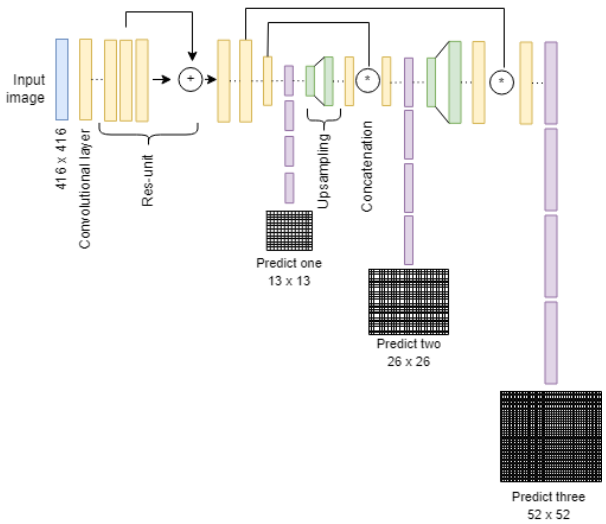

Figure 3. The framework of YOLOv3 neural network (based on Andriyanov et al. (2022))

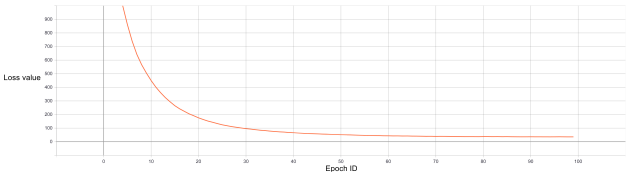

Figure 4. Loss function in relation to the number of epochs

2.3 Vehicle detection

Before performing car detection on the set of images, it is necessary to prepare the neural network model. It is important that the trained model allows for the correct detection of objects in images that were not previously used in the training process. Therefore, at the initial stage of data preparation, photos obtained on 30th August 2018, using another UAV, were used. The database included 50 images taken from a height of 100 m with a Flytech's Birdie unmanned airframe. The UAV was equipped with a non-metric Sony a6000 consumer-grade camera, with image resolution of 6000x4000 pixels. The flight area covered the same road, but the photos were characterized by different parameters resulting from a different camera model, time of day in which the flight was performed, higher flight speed, and different shutter type. This set of photos was chosen for the network training as it was necessary to check whether the prepared model will work correctly even in the case of using different equipment and flight parameters , such as altitude or speed. Several to several dozen cars were captured in each image, so despite having only 50 photos, the number of samples was much greater.

The image database was divided into two groups: training (70%) and test (30%). The training samples varied in terms of colour and rotation of the objects. The detector was trained for 100 epochs. The entire training process took over 3 hours and a mean Average Precision (mAP) value of 83.755% was obtained. It could be possible to increase the precision by using more samples (typically several thousand samples are used in object detection studies). From a certain point, the loss function values become constant so further increasing the number of epochs would not improve the result (Figure 4). However, the aim of this research is not to obtain the best possible prediction accuracy, but to assess the impact of car masking on photogrammetric products, so it was decided that such a value would be sufficient.

Figure 5. Original prediction converted into a binary mask image

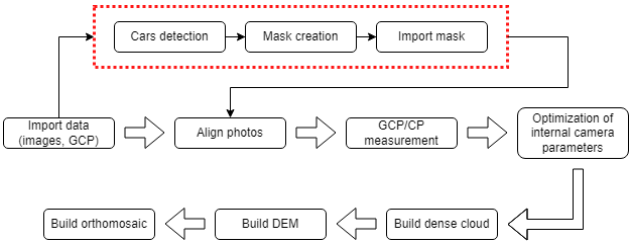

Figure 6. Processing stages

The next stage was the detection of cars in the main dataset with the use of the created model. In its predefined form, detected objects are presented within rectangles assigned with a class description and prediction accuracy. In order to enable using the results as masks for further photogrammetric processing, it was necessary to transform the original code in such a way that it would generate an artificial image with the same dimensions as the original image and would transform it into a binary form, where the background is 0 and the prediction regions are 1 (Figure 5). The detection process on the previously prepared model was quick, and for 216 photos with several to several dozen cars, it took several minutes.

2.4 Photogrammetric processing

The study was divided into two separate projects, differentiated only by the use or absence of masks resulting from automatic detection of cars in the images. The applied photogrammetric software uses the SfM-MVS algorithm. Due to the image acquisition with a non-metric camera, it was necessary to estimate the calibration parameters; they were treated as unknowns during bundle adjustment (self-calibration). Other parameters such as approximate exterior orientation parameters, focal length, and sensor size were loaded automatically based on EXIF files. The photogrammetric process consisted of several steps. In the first stage, preliminary photo alignment was performed. This process consists of feature detection, matching, and camera position estimation. All control points were then indicated on the images, 6 of them were selected as GCPs and the remaining 26 as check points used for accuracy analysis not involved in the alignment process. The projection centres were excluded from the process due to the low accuracy of the onboard navigation system. The next step was to perform camera parameter optimisation and to georeference the model through bundle adjustment. Then, photogrammetric products, such as dense point cloud, digital elevation model, and orthophotomap were generated. The processing workflow is presented in Figure 6.

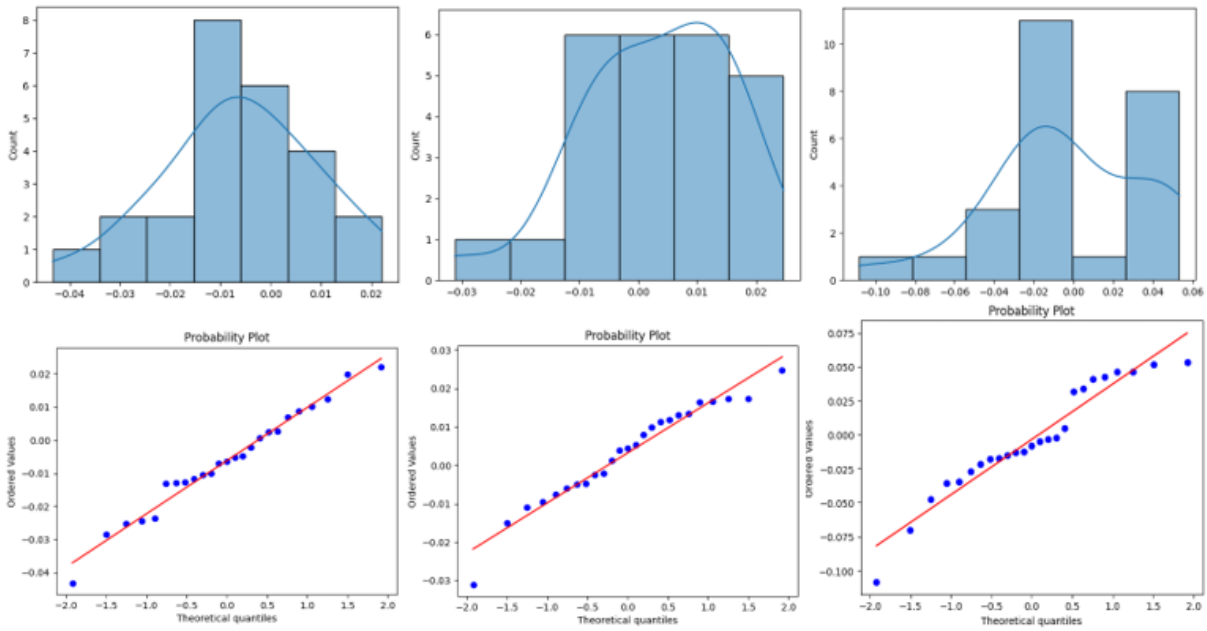

Figure 7. Histograms and Q-Q plots of deviations for X, Y, and Z coordinates

Table 1. RMSE values from accuracy assessment

RMSE X [m]	RMSE Y [m]	RMSE Z [m]	Total RMSE [m]
0.021448	0.012693	0.040035	0.047158

Table 2. p - values of Shapiro-Wilk normality testing

X deviations	Y deviations	Z deviations
0.902	0.380	0.106

3 Results

3.1 Accuracy assessment

Masking of vehicles detected by the YOLO algorithm changes the content of the source data. For control reasons, the processing pipeline was run twice: for original images and for masked images. All processes were performed on a computer with the following parameters: RAM 32 GB, Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz, GeForce GTX 1050 GPU. The number of obtained tie points resulting from the SfM algorithm was 223359 for the variant with masks and 221662 for the variant without masks. The alignment time was similar for both scenarios - around 11 minutes. The time necessary to generate the dense point cloud, DEM, and orthophotomap was also similar – 1 hour 10 minutes in total. The accuracy was assessed based on RMSE values calculated on check points from scenario with cars (Table 1).

In order to further analyse the dispersion of deviations for X, Y, and Z coordinates, the Shapiro-Wilk normality test was used. According to the test assumptions, distribution is normal when the p-value reaches values greater than 0.05 (Shapiro and Wilk, 1965). The test was performed using a Python script. The distribution was normal (Table 2). The distribution of deviations for X, Y, and Z coordinates presented in histograms and Q-Q plots is shown in (Figure 7).

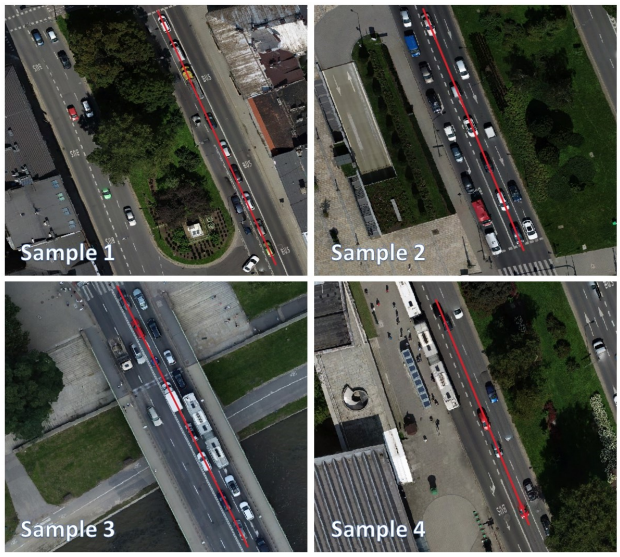

Figure 8. Samples for analysis with the places where the sections were made

3.2 Analysis of the generated products

The next stage constituted the main part of the research, i.e. the analysis of generated photogrammetric products. First, a comparative analysis of DEM was conducted. The generated DEMs were exported to the TIFF format and loaded into the QGIS program. In both scenarios, the products were created based on a dense point cloud. In both cases, there were approximately 34 million points in the point clouds . The generated DEMs had a size of 7949 x 13196 pixels and resolution of 6.9 cm/pix. Then, using the Profile Tool GitHub repository (2022) by Borys Jurgiel and Patrice Verchere, cross-sections of the terrain were made in the areas with visible cars. Samples were selected based on the orthophotomap generated for the variant without masks on which the cars were mapped. The locations of cross-sections are marked with a red line (Figure 8).

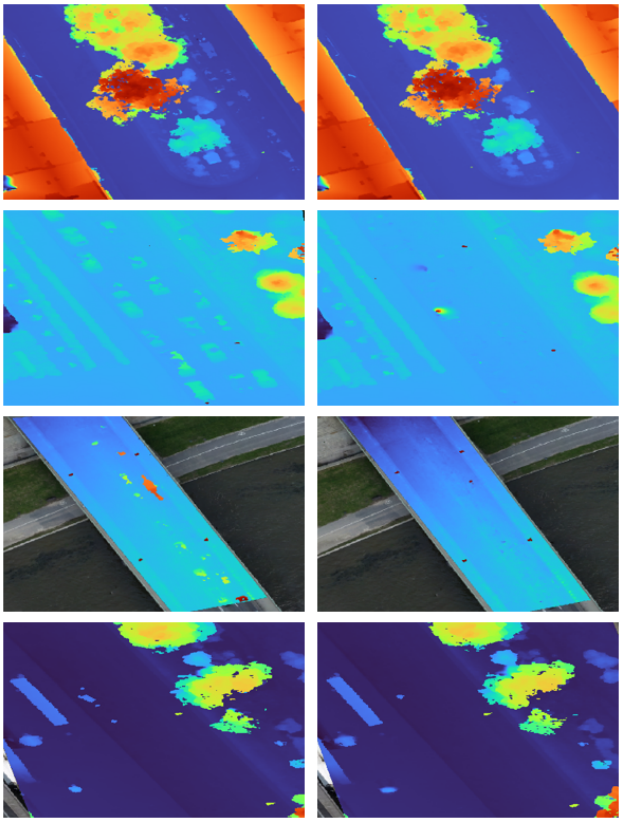

Figure 9. DEM at locations marked as samples. From the top: samples 1, 2, 3, 4, left without masking and right with masking

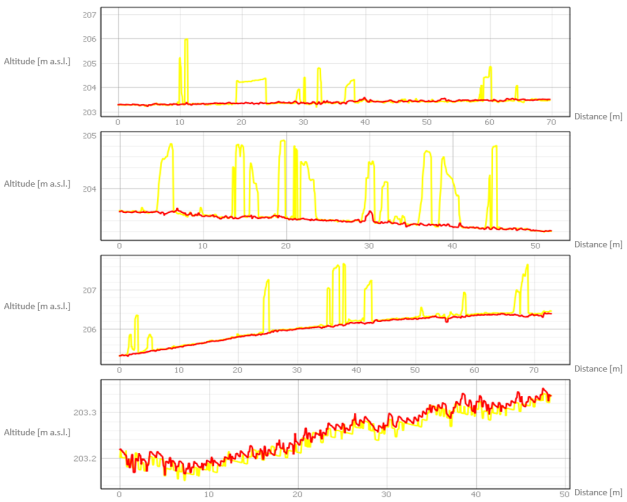

Figure 10. Cross-sections through the DEM made without masking the cars (yellow) and made with masks (red), from top: sample 1,2,3,4; x axis: distance in m, y axis: height above sea level

The choice of specific samples was based on the need to carry out the analysis in areas with the heaviest traffic, that is in the vicinity of crossroads and pedestrian crossings, which are included in samples 1-3. Sample 4 covers an area of different characteristics where the number of cars is much smaller and they are constantly in motion. A comparison of DEM grids for each of the samples in both scenarios is shown in Figure 9.

The resulting DEMs were visually assessed at sample locations. In all samples, the effect of car masking on the quality of the resulting product can be seen. DEMs based on masked images are noticeably smoother. In sample 1, the variant without masking makes the cars visible in the right-hand lane of the road. The variant with masking smoothed the street surface in this area. For sample 3, it was necessary to limit the DEM extent due to significant errors in the areas covered by water. The roadway area has become noticeably smoother and more consistent after masking. In sample 2, the errors can be seen on the DEM with masking because there were too few images without cars to obtain information about the area underneath them. The region covered by sample 2 was located at the beginning of the study area so the number of images covering this part was much smaller. Sample 4 presents a region with a small number of cars in motion. The result before and after masking for this area is similar. The graphs presented in Figure 10 show the course of the terrain line for each of the cross-sections in the samples. Variant without masks is marked in yellow, while the red colour indicates the variant with masks.

In the next stage, the generated orthophotomaps were compared. For both scenarios, the size of resulting products was 25244x42404 pixels with the resolution of 1.73 cm/pix. The obtained orthophotomaps were loaded into the QGIS program, where the effect of applying masks was visually assessed. Fragments of the orthophotomaps obtained with and without masks are shown in Figure 11. The cars moving in the course of taking successive images may cause the blur effect. Changing the position of an object makes it partially skipped or visible as a streak. This phenomenon is clearly visible on the orthophotomap generated without masking the cars. In a traditional form of processing, it is necessary to remove such artefacts manually by marking them into polygons and replacing them with areas extracted from a different image. The result obtained by automatically masking the cars shows that this process can be done fully automatically without operator intervention.

4 Discussion

The literature review shows that there is limited research on the masking of cars in images for photogrammetric processing of UAV data. Yang et al. conducted a study concerning automatic masking of moving cars (Yang et al., 2021). The research aimed to present a method for recognising and removing cars and combining it with 3D reconstruction, thus eliminating errors related to the geometry and texturing of cars within roads, which can be of utmost importance when creating smart cities. The results show that the proposed method mitigates problems occurring in urban scenes. High automation is an additional advantage. However, the paper analyses 3D modelling and texture overlay. The presented research extends previous findings by analysing the effect of masking on DEM and orthophotomap improvement for photogrammetric road measurements. Design and monitoring require reliable information concerning the area under study. The presence of cars in the images is partly eliminated by the RANSAC algorithm, but in the case of heavy traffic, there are residual errors in a dense point cloud and further products such as DEMs and orthophotomaps. Cars are not an integral part of the landscape so including them in photogrammetric products is unnecessary. In addition, when the location of objects changes between successive photographs, it causes the blurring effect, which is visible on the orthophotomap.

The results showed that masking the cars does not affect the processing accuracy. The number of obtained tie points also remains at a similar level, so it can be concluded that removing cars from the images does not disturb the process of feature search and matching algorithms. The processing time has not increased either. Masking, on the other hand, allows for the removal of unnecessary artefacts from a DEM, as shown in samples 1-3. In the case of less congested areas where single cars were visible, it is not necessary to mask them because they are removed by the SfM algorithm, as shown in sample 4. The problematic areas are those of heavy traffic and slow movement, for example in the vicinity of crossroads, traffic

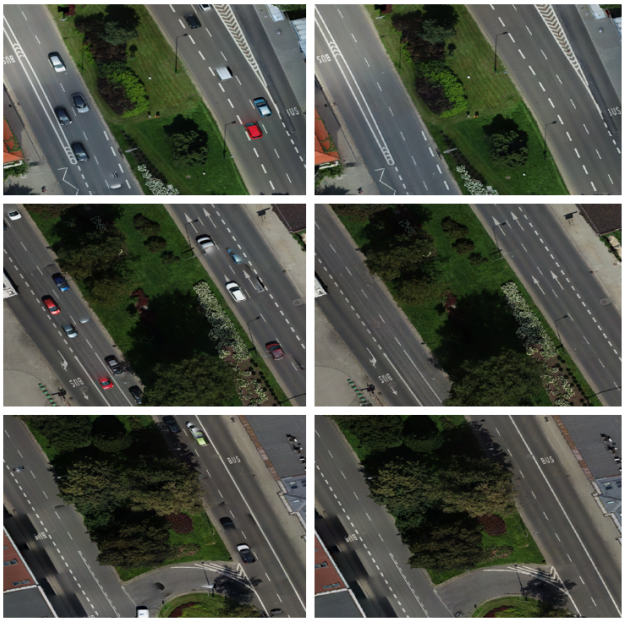

Figure 11. Comparison of the orthophotomaps generated without automatic masking (left) and with automatic masking (right)

lights, and pedestrian crossings. The analysis of the obtained orthophotomaps shows that the use of automatic masking allows for the complete elimination of the need to manually remove artefacts from the final product. The full automation and speed of detection allow for obtaining improved product quality without increasing the time required to process the data, which is a serious disadvantage of manual masking or artefact cleaning. The trained model was based on a small number of training and test images, future research may include a much larger number of samples to improve prediction accuracy. Additionally, only one class "cars" was considered in the algorithm. Other objects such as bicycles, motorbikes, or buses were not taken into account. In future research, it is possible to extend the obtained model to include additional object classes.

5 Conclusions

The research aimed to demonstrate that automatic masking of cars in images acquired with UAVs can positively affect the quality of photogrammetric products. A digital elevation model generated from a dense point cloud and an orthophotomap were analysed. To achieve the research objectives, automatic car detection was performed using the state-of-the-art object detection YOLO v3 algorithm, and the results were then processed into masks. The data prepared in this way were subjected to photogrammetric processing in two scenarios: with and without masks covering the cars. The accuracy analysis was based on the RMS errors of the coordinates read on check points in both scenarios. In addition, Shapiro-Wilk normality tests were performed. The accuracy assessment shows that masking cars in the images has no effect on the final accuracy. This is due to the small number of incorrectly designated tie points in relation to their total number. The main part of the study was a comparative analysis of photogrammetric products generated from both scenarios. The analysis of DEM and orthophotomap shows that masking the cars can improve the quality of the final products. Performing photogrammetric flights over busy roads involves capturing numerous cars, which are then transferred as erroneous fragments to photogrammetric products. By using masking, it is possible to eliminate their influence at the beginning of the alignment without the need to manually interfere with the resulting products. In addition, the use of automatic detection significantly reduces the time required for processing. The trained network can be useful not only in the project from which the images were acquired for training but also in studies done with a different UAV, camera, flight altitude, and ground sample distance. The model obtained in the study was trained using a small database of images from another flight (50 images). Despite the small test and training database, detection accuracy of 83 % was achieved, which is sufficient to highlight the differences between the process without and with masks.

Acknowledgments

The author would like to thank the authors of the open-source YOLO algorithm used in this research.

References

Andriyanov, N., Khasanshin, I., Utkin, D., Gataullin, T., Ignar, S., Shumaev, V., and Soloviev, V. (2022). Intelligent system for estimation of the spatial position of apples based on YOLOv3 and real sense depth camera d415. *Symmetry*, 14(1):148, doi:10.3390/sym14010148.

Bao, W., Ren, Y., Wang, N., Hu, G., and Yang, X. (2021). Detection of abnormal vibration dampers on transmission lines in UAV remote sensing images with PMA-YOLO. *Remote Sensing*, 13(20):4134, doi:10.3390/rs13204134.

Bianco, S., Ciocca, G., and Marelli, D. (2018). Evaluating the performance of structure from motion pipelines. *Journal of Imaging*, 4(8):98, doi:10.3390/jimaging4080098.

Campana, S. (2017). Drones in archaeology. State-of-the-art and future perspectives. *International Journal of Archaeological Prospection*, 24(4):275—-296, doi:10.1002/arp.1569.

Cardenal, J., Fernández, T., Pérez-García, J. L., and Gómez-López, J. M. (2019). Measurement of road surface deformation using images captured from UAVs. *Remote Sensing*, 11(12):1507, doi:10.3390/rs11121507.

Carrera-Hernández, J., Levresse, G., and Lacan, P. (2020). Is UAV-SfM surveying ready to replace traditional surveying techniques? *International journal of remote sensing*, 41(12):4820–4837, doi:10.1080/01431161.2020.1727049.

Coombs, C., Hislop, D., Taneva, S. K., and Barnard, S. (2020). The strategic impacts of intelligent automation for knowledge and service work: An interdisciplinary review. *The Journal of Strategic Information Systems*, 29(4):101600, doi:10.1016/j.jsis.2020.101600.

Dainelli, R., Toscano, P., Di Gennaro, S. F., and Matese, A. (2021). Recent advances in unmanned aerial vehicles forest remote sensing — a systematic review. part i: A general framework. *Forests*, 12(3):327, doi:10.3390/f12030327.

Delavarpour, N., Koparan, C., Nowatzki, J., Bajwa, S., and Sun, X. (2021). A technical study on UAV characteristics for precision agriculture applications and associated practical challenges. *Remote Sensing*, 13(6):1204, doi:10.3390/rs13061204.

Eltner, A. and Sofia, G. (2020). Structure from motion photogrammetric technique. In *Developments in Earth surface processes*, volume 23, pages 1–24. Elsevier, doi:10.1016/B978-0-444-64177-9.00001-1.

Fiz, J. I., Martín, P. M., Cuesta, R., Subías, E., Codina, D., and Cartes, A. (2022). Examples and results of aerial photogrammetry in archeology with UAV: Geometric documentation, high resolution multispectral analysis, models and 3D printing. *Drones*, 6(3):59, doi:10.3390/drones6030059.

Ge, L., Li, X., and Ng, A. H.-M. (2016). UAV for mining applications: A case study at an open-cut mine and a longwall mine in New South Wales, Australia. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5422–5425. IEEE,

doi:10.1109/IGARSS.2016.7730412.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA*, pages 580–587. doi:10.1109/CVPR.2014.81.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2015). Fast R-CNN. In *the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015*, pages 7–13. doi:10.1109/ICCV.2015.169.

Gromada, K., Siemiątkowska, B., Stecz, W., Płochocki, K., and Woźniak, K. (2022). Real-time object detection and classification by UAV equipped with SAR. *Sensors*, 22(5):2068, doi:10.3390/s22052068.

Gruen, A. (2021). Everything moves: The rapid changes in photogrammetry and remote sensing. *Geo-spatial Information Science*, 24(1):33–49, doi:10.1080/10095020.2020.1868275.

Han, X., Chang, J., and Wang, K. (2021). Real-time object detection based on YOLO-v2 for tiny vehicle object. *Procedia Computer Science*, 183:61–72, doi:10.1016/j.procs.2021.02.031.

Horzyk, A. and Ergün, E. (2020). YOLOv3 precision improvement by the weighted centers of confidence selection. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, doi:10.1109/IJCNN48605.2020.9206848.

Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J., and Rosette, J. (2019). Structure from motion photogrammetry in forestry: A review. *Current Forestry Reports*, 5(3):155–168, doi:10.1007/s40725-019-00094-3.

Indolia, S., Goswami, A. K., Mishra, S. P., and Asopa, P. (2018). Conceptual understanding of convolutional neural network-a deep learning approach. *Procedia computer science*, 132:679–688, doi:10.1016/j.procs.2018.05.069.

Jiang, S., Jiang, C., and Jiang, W. (2020). Efficient structure from motion for large-scale UAV images: A review and a comparison of SfM tools. *ISPRS Journal of Photogrammetry and Remote Sensing*, 167:230–251, doi:10.1016/j.isprsjprs.2020.04.016.

Ju, M., Luo, H., Wang, Z., Hui, B., and Chang, Z. (2019). The application of improved YOLO v3 in multi-scale target detection. *Applied Sciences*, 9(18):3775, doi:10.3390/app9183775.

Jurgiel, B. and Verchere, P. (2022). Profile Tool GitHub repository. Available online: `https://github.com/etiennesky/profiletool`, Last accessed April 2022.

Kaivosoja, J., Hautsalo, J., Heikkinen, J., Hiltunen, L., Ruuttunen, P., Näsi, R., Niemeläinen, O., Lemsalu, M., Honkavaara, E., and Salonen, J. (2021). Reference measurements in developing UAV systems for detecting pests, weeds, and diseases. *Remote Sensing*, 13(7):1238, doi:10.3390/rs13071238.

Koay, H. V., Chuah, J. H., Chow, C.-O., Chang, Y.-L., and Yong, K. K. (2021). YOLO-RTUAV: Towards real-time vehicle detection through aerial images with low-cost edge devices. *Remote Sensing*, 13(21):4196, doi:10.3390/rs13214196.

Koeva, M., Muneza, M., Gevaert, C., Gerke, M., and Nex, F. (2018). Using UAVs for map creation and updating. a case study in Rwanda. *Survey Review*, 50(361):312–325, doi:10.1080/00396265.2016.1268756.

Li, C.-Y. and Lin, H.-Y. (2020). Vehicle detection and classification in aerial images using convolutional neural networks. In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, Valletta, Malta*, volume 5, pages 775–782. doi:10.5220/0008941707750782.

Li, Y. and Liu, C. (2019). Applications of multirotor drone technologies in construction management. *International Journal of Construction Management*, 19(5):401–412, doi:10.1080/15623599.2018.1452101.

Luo, X., Tian, X., Zhang, H., Hou, W., Leng, G., Xu, W., Jia, H., He, X., Wang, M., and Zhang, J. (2020). Fast automatic vehicle detection in UAV images using convolutional neural networks. *Remote Sensing*, 12(12):1994, doi:10.3390/rs12121994.

Nyimbili, P. H., Demirel, H., Seker, D., and Erden, T. (2016). Structure from motion (SfM) – approaches and applications. In *Proceedings of the international scientific conference on applied sciences, Antalya, Turkey*, pages 27–30.

Park, S. and Choi, Y. (2020). Applications of unmanned aerial vehicles in mining from exploration to reclamation: A review. *Minerals*, 10(8):663, doi:10.3390/min10080663.

Pessacg, F., Gómez-Fernández, F., Nitsche, M., Chamo, N., Torrella, S., Ginzburg, R., and De Cristóforis, P. (2022). Simplifying UAV-based photogrammetry in forestry: How to generate accurate digital terrain model and assess flight mission settings. *Forests*, 13(2):173, doi:10.3390/f13020173.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016*, pages 779–788. doi:10.48550/arXiv.1506.02640.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, doi:10.48550/arXiv.1804.02767.

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, doi:10.1109/TPAMI.2016.2577031.

Roberts, R., Inzerillo, L., and Di Mino, G. (2020). Using UAV based 3D modelling to provide smart monitoring of road pavement conditions. *Information*, 11(12):568, doi:10.3390/info11120568.

Sahin, O. and Ozer, S. (2021). Yolodrone: Improved yolo architecture for object detection in drone images. In *2021 44th International Conference on Telecommunications and Signal Processing (TSP)*, pages 361–365. IEEE, doi:10.1109/TSP52935.2021.9522653.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, doi:10.2307/2333709.

Snavely, N., Seitz, S. M., and Szeliski, R. (2008). Modeling the world from internet photo collections. *International journal of computer vision*, 80(2):189–210, doi:10.1007/s11263-007-0107-3.

Tan, L., Lv, X., Lian, X., and Wang, G. (2021). YOLOv4_Drone: UAV image target detection based on an improved YOLOv4 algorithm. *Computers & Electrical Engineering*, 93:107261, doi:10.1016/j.compeleceng.2021.107261.

Tan, Y. and Li, Y. (2019). UAV photogrammetry-based 3D road distress detection. *ISPRS International Journal of Geo-Information*, 8(9):409, doi:10.3390/ijgi8090409.

Wang, J., Su, S., Wang, W., Chu, C., Jiang, L., and Ji, Y. (2022). An object detection model for paint surface detection based on improved yolov3. *Machines*, 10(4):261, doi:10.3390/machines10040261.

Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., and Lan, X. (2020). A review of object detection based on deep learning. *Multimedia Tools and Applications*, 79(33):23729–23791, doi:10.1007/s11042-020-08976-6.

Xu, Z.-F., Jia, R.-S., Sun, H.-M., Liu, Q.-M., and Cui, Z. (2020). Light-YOLOv3: fast method for detecting green mangoes in complex scenes using picking robots. *Applied Intelligence*, 50(12):4670–4687, doi:10.1007/s10489-020-01818-w.

Yahya, M. Y., Shun, W. P., Yassin, A. M., and Omar, R. (2021). The challenges of drone application in the construction industry. *Journal of Technology Management and Business*, 8(1):20–27, doi:10.30880/jtmb.2021.08.01.003.

Yang, C., Zhang, F., Gao, Y., Mao, Z., Li, L., and Huang, X. (2021). Moving car recognition and removal for 3D urban modelling using oblique images. *Remote Sensing*, 13(17):3458, doi:10.3390/rs13173458.

Zhao, Z.-Q., Zheng, P., Xu, S.-t., and Wu, X. (2019). Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232,

doi:10.1109/TNNLS.2018.2876865.

Zhu, Q., Shang, Q., Hu, H., Yu, H., and Zhong, R. (2021). Structure-aware completion of photogrammetric meshes in urban road environment. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175:56–70, doi:10.1016/j.isprsjprs.2021.02.010.

Zulkipli, M. A. and Tahar, K. N. (2018). Multirotor UAV-based photogrammetric mapping for road design. *International Journal of Optics*, 2018:7, doi:10.1155/2018/1871058.

Šafář, V., Potůčková, M., Karas, J., Tlustý, J., Štefanová, E., Jančovič, M., and Cígler Žofková, D. (2021). The use of UAV in cadastral mapping of the Czech Republic. *ISPRS International Journal of Geo-Information*, 10(6):380, doi:10.3390/ijgi10060380.