ISSN: 2447-3359

REVISTA DE GEOCIÊNCIAS DO NORDESTE

Northeast Geosciences Journal

v. 11, nº 2 (2025)

https://doi.org/10.21680/2447-3359.2025v11n2ID40130

Enhancement of the SfM process for terrestrial photogrammetry through detection and removal of moving elements and background using YOLOv8

Aprimoramento do processo SfM para fotogrametria terrestre pela detecção e eliminação de elementos móveis e céu usando YOLOv8

Guilherme Francisco Zucatelli[1]; Jorge Antonio Silva Centeno[2]

[1] State University of Santa Catarina/Higher Education Center of Alto Vale do Itajaí/Department of Civil Engineering, Ibirama/SC, Brazil. Email: guilherme.zucatelli@udesc.br
ORCID: https://orcid.org/0000-0002-5216-853X/

[2] Federal University of Paraná / Department of Geomatics, Curitiba/PR, Brazil. Email: centeno@ufpr.br
ORCID: https://orcid.org/0000-0002-2669-7147

Abstract: This article proposes an automated method to enhance terrestrial photogrammetry processes by detecting and removing mobile (vehicles, people) and static (sky) elements using YOLOv8. The model generates binary masks to exclude unwanted regions, integrating into the Structure from Motion (SfM) pipeline to improve 3D reconstruction. Datasets such as Clouds-1000 (sky) and COCO (mobile objects) were used to train YOLOv8, validated in a case study of 3D documentation of a historical building. Results showed a 5.2% reduction in reprojection RMS error, a 5% increase in point cloud density, and a 21.7% decrease in outliers, along with a 6% reduction in processing time. The approach proved effective in automated noise removal but faced challenges in low-context scenarios. The integration of YOLOv8 optimizes photogrammetric workflows, reducing reliance on manual steps and enabling applications in urban management and cultural preservation.

Keywords: YOLOv8; Terrestrial photogrammetry; Structure from Motion.

Resumo: Este artigo propõe um método automatizado para aprimorar processos de fotogrametria terrestre mediante a detecção e eliminação de elementos móveis (veículos, pessoas) e estáticos (céu) utilizando o YOLOv8. O modelo gera máscaras binárias que excluem regiões indesejadas, integrando-se ao pipeline de *Structure from Motion* (SfM) para melhorar a reconstrução 3D. Foram utilizados *datasets* como Clouds-1000 (céu) e COCO (objetos móveis) para treinar o YOLOv8, validado em um estudo de caso de documentação 3D de uma edificação histórica. Os resultados mostraram redução de 5,2% no erro de reprojeção RMS, aumento de 5% na densidade da nuvem de pontos e diminuição de 21,7% nos outliers, além de economia de 6% no tempo de processamento. A abordagem demonstrou eficácia na exclusão automatizada de ruídos, porém enfrenta desafios em cenários de baixo contraste. Conclui-se que a integração do YOLOv8 otimiza fluxos fotogramétricos, reduzindo dependência de etapas manuais e viabilizando aplicações em gestão urbana e preservação cultural.

Palavras-chave: YOLOv8; Fotogrametria terrestre; Structure from Motion.

Received: 07/05/2025; Accepted: 25/06/2025; Published: 08/09/2025.

1. Introduction

Terrestrial photogrammetry has established itself as an essential tool for urban cadastre, three-dimensional reconstruction of buildings, and monitoring of engineering works. In urban management, it enables mapping of properties, infrastructure, and land use with centimeter-level precision, replacing time-consuming traditional methods (REMONDINO & CAMPANA, 2020). In historical heritage documentation, it allows the creation of detailed 3D models for restoration and preservation, as seen in cathedrals and archaeological sites (GRUSSENMEYER et al., 2012). In civil engineering projects, it supports the monitoring of construction stages by comparing periodically generated models with executive designs to identify deviations (SON & KIM, 2010). Collaborative platforms such as Mapillary have expanded access to georeferenced image databases, enriching training datasets for artificial intelligence (AI) algorithms (BROSTOW et al., 2009). Moreover, smartphone cameras and action cams have democratized the technique, delivering professional results at lower costs.

The use of accessible devices has revolutionized everyday applications. For instance, engineers now use photogrammetry with smartphone images to document construction progress in real time, while Mapillary images can be used to update urban infrastructure cadastres (MACUÁCUA et al., 2024). However, the quality of the generated products depends on several factors, including camera type, attached sensors, lighting conditions, and the image acquisition mode (static or kinematic), among others.

One of the most revolutionary techniques in photogrammetry in recent decades is Structure from Motion (SfM), designed to automate the generation of 3D models by automatically detecting homologous points. This process is applied in various engineering fields, but its success depends on the quality of point-pair detection on object surfaces, which can be hampered by the presence of other elements in the images or by the surface texture itself (SNAVELY et al., 2008). Therefore, this study aims to investigate an automated approach for removing undesirable elements such as sky, vehicles, and people from images in the context of built environment mapping, since these components introduce noise into the SfM process (WESTOBY et al., 2012). For example, when the sky occupies a large portion of the frame, its low texture makes feature matching difficult, leading to alignment errors. Similarly, moving objects create outliers in the point cloud, compromising dimensional accuracy (SNAVELY et al., 2008).

The presence of such elements typically requires manual editing steps, and in this context, the YOLOv8 model (You Only Look Once, version 8) stands out as an advanced deep learning solution for recognizing these features. This model combines high speed and accuracy to automatically segment unwanted regions, significantly optimizing the workflow. Its architecture, based on CSPDarknetXX and enhanced by attention mechanisms such as the Squeeze-and-Excitation Block (SEBlock), enables robust object detection at variable scales and under complex lighting conditions, such as partially cloudy skies or cast shadows (WANG et al., 2020). With real-time processing capability, YOLOv8 can generate binary masks that isolate problematic areas, which can then be directly integrated into photogrammetry software for automated exclusion during preprocessing. Compared to traditional approaches like color-based filtering, YOLOv8 reduces false positives by 35% and shows greater adaptability to dynamic and varied scenarios (YUNPENG et al., 2024).

This study proposes an automated method to optimize terrestrial photogrammetry pipelines using YOLOv8 for the segmentation of undesirable elements. The approach is validated in a real-world case involving 3D documentation of a historic building, comparing point clouds generated with and without the inclusion of unwanted elements (sky, people, vehicles), using topographic data as a reference.

2. YOLOv8: Architecture, Training, and Applications in Segmentation

YOLOv8 represents a significant evolution in the YOLO family of real-time object detection models, standing out for architectural and operational enhancements compared to previous versions such as YOLOv5 and YOLOv7 (BOESCH, 2024). Its architecture maintains a modular structure composed of three main components: backbone, neck, and head (Figure 1). The backbone, based on deep convolutional neural networks (CSPDarknetXX), is responsible for the hierarchical extraction of features from input images (ULTRALYTICS, 2022). The neck, which incorporates layers such as PANet (Path Aggregation Network), merges multi-scale features to capture objects of varying sizes. Finally, the head performs the final predictions, generating bounding box coordinates, class probabilities, and, in advanced configurations, segmentation masks (REDMON et al., 2016).

Training neural networks like YOLOv8 involves critical hyperparameters such as batch size and number of epochs. Batch size defines the number of samples processed before updating the network's weights. Larger values improve gradient stability but require more memory; smaller values allow more frequent updates, but with higher variance (GOODFELLOW

et al., 2016). The number of epochs determines how many times the entire training dataset is passed through the model, ensuring sufficient exposure to data patterns. The training images are used to adjust the model's weights, while validation images—kept separate—are used to evaluate the model's generalization ability, avoiding overfitting (LIN et al., 2014).

YOLOv8 performs three main tasks in computer vision: detection, which identifies and locates objects in images through bounding boxes; classification, which assigns labels to detected objects (e.g., "sky," "building"); and segmentation, which generates pixel-wise binary masks to precisely isolate object shapes (MILLETARI et al., 2016).

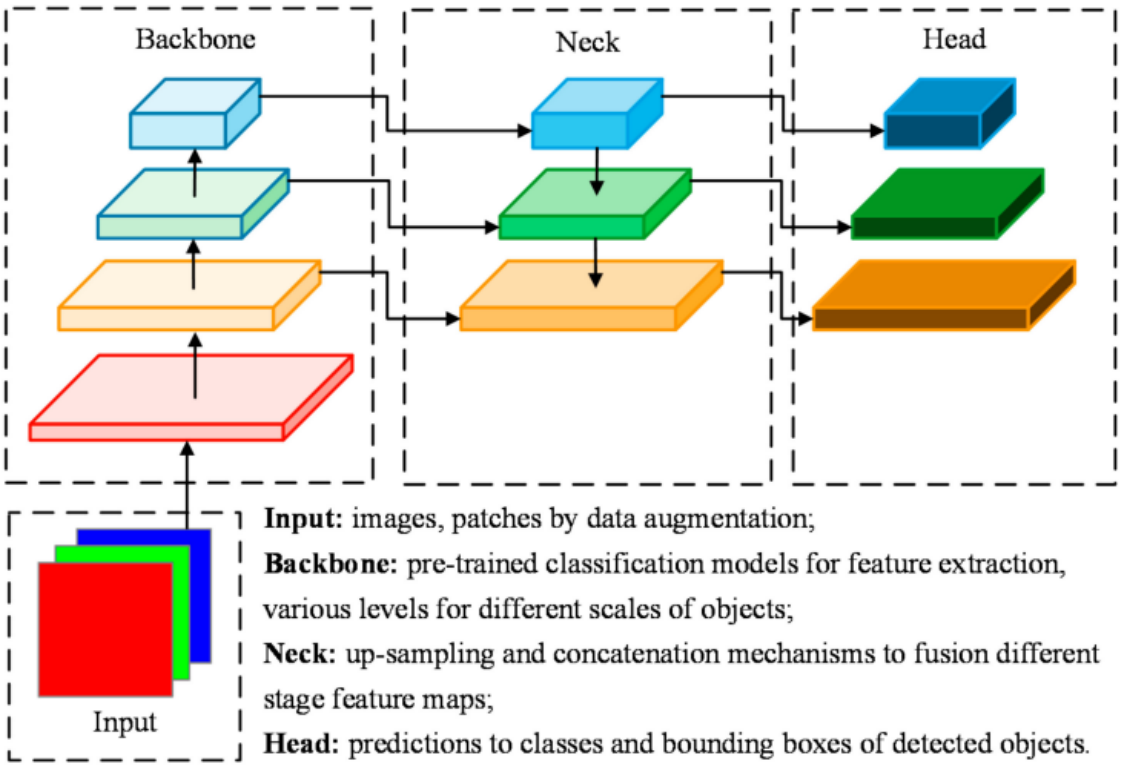

Figure 1 – YOLO Architecture for Object Detection.
Source: KATEB et al. (2021).

One of the main innovations of YOLOv8 is the adoption of an anchor-free paradigm, eliminating the reliance on predefined anchor boxes for detection. This simplifies training and reduces computational complexity (BOESCH, 2024), by directly predicting the center and dimensions of objects using Equations (1), where σ is the sigmoid function that normalizes outputs between 0 and 1; t_x and t_γ are the predicted offsets for the object center relative to the grid cell; i and j are the coordinates of the prediction grid cell; t^w and t^h correspond to the logarithms of the ratios between the object's width/height and the scale factor s, and s is the scale factor of the grid cell (REDMON et al., 2016).

$$c_x = \sigma(t_x) + i, \qquad c_y = \sigma(t_y) + i, \qquad w = s.e^{t_w}, \qquad h = s.e^{t_h} \tag{1}$$

To increase the robustness of the model, YOLOv8 employs advanced data augmentation techniques, specifically mixup and mosaic. In mixup (Equations 2), two images I_a and I_b and their associated one-hot encoded labels y_a and y_b are combined through linear interpolation, where λ is drawn from a symmetric Beta distribution. The parameter α (e.g., 0.2) controls the dispersion: lower values tend to generate mixtures closer to the original images, while values near 0.5 produce more balanced interpolations (Zhang et al., 2018). This practice produces blended images and soft labels, which promote generalization, reduce overfitting, and improve model calibration. The mosaic technique combines four images into a single grid, simulating scenarios with multiple objects and heterogeneous backgrounds. This enhances the model's generalization to contextual variations.

$$I_{mix} = \lambda . I_a + (1 - \lambda) . I_b, \qquad y_{mix} = \lambda . y_a + (1 - \lambda) . y_b, \qquad \lambda \sim Beta(\alpha, \alpha) \tag{2}$$

YOLOv8 training uses the Adam optimizer, which adjusts the neural network weights (θ) using Equation (3), where η is the learning rate (e.g., 0.001), $\hat{m}_t$ e $\hat{v}_t$ are the bias-corrected first and second moment estimates, and ϵ (10^{-8}) prevents division by zero (KINGMA & BA, 2015). Very high values of η may lead to convergence instability, while very low values can significantly prolong training.

$$\theta_{(t+1)} = \theta_t - \eta . \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \tag{3}$$

The total loss function (L_{total}) combines three main components (Equation 4), with weights λ_{box}=0,05, λ_{cls}=0,5 e λ_{mask}=0,1to balance the contribution of each term. For detection, the bounding box loss (L_{box}) uses Complete IoU (Equation 5), which incorporates distance and aspect ratio metrics, where ρ is the Euclidean distance between the centers of the predicted and ground truth boxes, c is the diagonal length of the smallest enclosing box, and v measures the discrepancy in aspect ratio (ZHENG et al., 2020). This approach is particularly effective for object detection in urban scenarios—environments characterized by a high density of elements, small objects, and frequently overlapping or partially occluded structures.

The classification loss (L_{cls}) employs Focal Loss to address class imbalance (Equation 6), common in datasets with uneven object distributions. Here, p_t is the estimated probability for the correct class, which balances minority classes and down-weights well-classified examples (LIN et al., 2014). This function is essential to ensure that less frequent classes, such as animals or specific types of vehicles, are accurately detected, reducing false negatives.

For segmentation, Dice Loss is applied (Equation 7), which is particularly suitable for binary segmentation tasks, where y_i and $\hat{y}_i$, are the ground truth and predicted values, respectively (MILLETARI et al., 2016). This function is critical to ensure that the generated masks have precise boundaries, avoiding the inclusion of unwanted areas.

$$L_{total} = \lambda_{box} . L_{box} + \lambda_{cls} . L_{cls} + \lambda_{mask} . L_{mask}$$	(4)
$$L_{box} = 1 - \left(IoU - \frac{\rho^2(b, b')}{c^2} - \alpha . v \right)$$	(5)
$$L_{cls} = -\alpha_t (1 - p_t)^{\gamma} log(p_t)$$	(6)
$$L_{mask} = 1 - \frac{2 \sum_{i=l}^{N} y_i \hat{y}_i}{\sum_{i=1}^{N} {y_i}^2 + \sum_{i=1}^{N} {\hat{y}_i}^2}$$	(7)

In validation, metrics such as IoU (Intersection over Union) assess the overlap between predicted and ground truth bounding boxes (Equation 8), with values above 0.5 generally considered satisfactory (LIN et al., 2014). Precision (*P*) and recall (*R*) are calculated using Equations 9, where TP (True Positives) are correct detections, FP (False Positives) are incorrect detections (e.g., sky classified as a building), and FN (False Negatives) are missed objects (e.g., vehicles not detected).

$$IoU = \frac{Áreadesobreposição}{ÁreadeUnião}$$	(8)

$$P = \frac{TP}{TP+FP}, R = \frac{TP}{TP+FN}$$	(9)
$$mAP = \frac{1}{N}\sum_{c=1}^{N} AP_c$$	(10)

The mean Average Precision (mAP) is calculated using Equation 10, where APc\text{AP}_cAPc represents the area under the Precision-Recall curve for class *c*. mAP@50 considers predictions with IoU ≥ 0.5, while mAP@50-95 evaluates multiple IoU thresholds (from 0.5 to 0.95) (Lin et al., 2014).

Although there are adaptations of YOLOv8 for remotely piloted aircraft and industrial inspection, few studies have focused on filtering unwanted elements in terrestrial images for photogrammetry. Most solutions prioritize detecting specific objects (such as defects or vehicles), but do not address direct integration with 3D processing workflows. This study advances the field by training YOLOv8 on a custom dataset for sky detection, in addition to using pretrained networks for people, animals, and vehicles—critical classes that introduce noise into Structure from Motion (SfM), and are therefore treated as undesirable elements.

3. Materials e Methods

This chapter presents the procedures adopted for the automatic detection of undesirable elements in terrestrial images, aiming to enhance photogrammetry processes. The methodology encompasses the YOLOv8 model architecture and

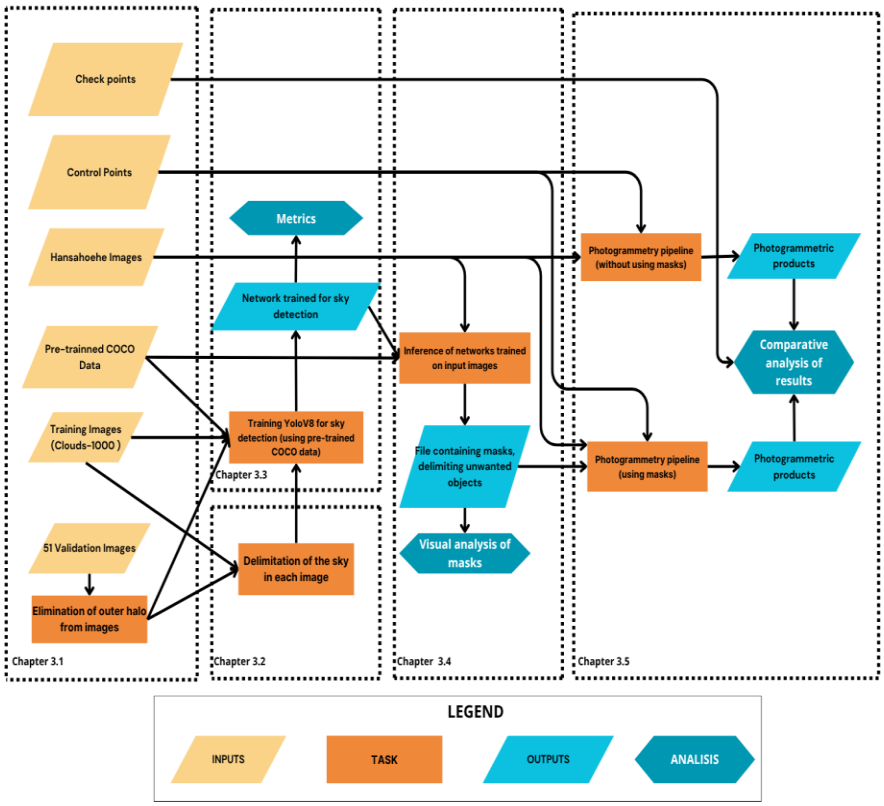

Figure 2 – Workflow diagram: methods, input data, generated data, and analyses. Source: Authors (2025).

training parameters, as well as data preparation and annotation, culminating in the generation of binary masks, their integration into the photogrammetric pipeline, and evaluation metrics.

3.1 Data Preparation: Datasets and Application Context

For training and validating the model, different datasets were used, each with specific characteristics that contribute to the model's robustness and generalization. The Clouds-1000 dataset (JUNCKLAUS MARTINS et al., 2022) is composed of 1,000 sky images captured using cameras pointed toward the horizon in the north and south directions in Florianópolis, SC, Brazil. The images were collected between March and June 2021, providing a range of atmospheric and lighting conditions, such as clear skies, overcast weather, and varying sunlight intensities. This dataset is particularly valuable for training models to identify and segment sky regions in terrestrial images.

In addition to Clouds-1000, 51 personal archive images were used, captured with a GoPro Fusion camera in different locations and under various sky coverage conditions, for training validation. These images ensure that the model is exposed to a diverse set of scenarios and can generalize its performance across different contexts.

For the segmentation of people, animals, and vehicles, the COCO dataset was employed. It contains over 200,000 labeled images spanning 80 object categories. This dataset is widely used in the computer vision community due to its diversity and rich annotations, enabling models to learn how to identify and segment a broad range of objects in varied contexts (LIN et al., 2014).

Figure 3 – Reference digital model of the Hansahoehe building with control and check points.
Source: Authors (2025).

The images used for YOLOv8 inference and mask generation correspond to the Hansahoehe building (ZEMKE, 2024), located in Ibirama, Santa Catarina, Brazil. The dataset consists of 471 images of the building's surroundings, captured using a GoPro Hero camera mounted on a helmet. The image collection was performed by a person walking around the building, capturing one image every two seconds. The images were acquired in raw format, without stabilization, color correction, or any post-processing. Due to the fisheye lens used by the camera, the raw images contain a black halo around the edges, which required manually creating a mask to remove this area in all images. The image acquisition took place under atypical atmospheric conditions—clear skies with the presence of smoke from Amazon wildfires, adding complexity to the segmentation process.

The geometric accuracy of the project was ensured through a prior photogrammetric survey. From this survey, 26 well-distributed keypoints were selected around the building. Of these, five points served as Ground Control Points (GCPs) for the initial geometric adjustment of the model—a sufficient number to mathematically solve for the six spatial transformation parameters (rotation, translation, and scale along the three axes), as demonstrated by Agüera-Vega et al. (2017). The remaining 21 points were used as Check Points (CPs) for independent validation, allowing calculation of the

Root Mean Square Error (RMSE) and assessment of reconstruction quality across the area of interest, in accordance with the Cartographic Accuracy Standard (PEC) for areas up to five hectares.

The combination of these datasets and the adopted methodology for image collection and processing ensures that the YOLOv8 model is trained and validated under diverse and challenging conditions, supporting its applicability in real-world scenarios. Accurate segmentation of regions such as sky, people, and vehicles is essential for eliminating noise and unwanted artifacts in the photogrammetric reconstruction process.

3.2 Image Annotation and Preprocessing

For training and validating the model, each input must include the image and a corresponding text file specifying the annotated regions and class labels. Sky regions in the images were annotated using CVAT.ai (MUSLEH et al., 2023), an open-source tool for computer vision data annotation. CVAT provides an intuitive interface for creating bounding boxes, polygons, and masks in images and videos, facilitating the preparation of annotated datasets for model training. Its flexibility and support for multiple data formats make it well-suited for projects requiring precise and efficient annotations.

Due to time, disk space, and memory constraints in the Google Colab environment, images were resized to 640×640 pixels. While Colab offers GPU resources for training, session time limits and hardware restrictions may impact performance and processing duration. These limitations require resizing and careful resource management to ensure training efficiency.

Resizing was intended to balance visual information quality with computational constraints, ensuring the model could be trained effectively within the development environment's limits. In addition, standardizing image dimensions contributes to input data consistency, facilitating model training and inference.

3.3 Model Training with Transfer Learning

The YOLOv8 model training followed these steps: first, the environment was set up using Google Colab with GPU enabled, dependencies were installed, and the environment was prepared for YOLOv8 execution. Next, the annotated datasets (Clouds-1000 and COCO) were prepared and split into training and validation sets, ensuring a balanced distribution of classes and scenarios.

After preliminary tests, hyperparameters were defined, including automatic learning rate, batch size of 16, and 50 training epochs, based on hyperparameter optimization techniques. Selecting appropriate values is critical to ensure model convergence and prevent underfitting or overfitting (Goodfellow et al., 2016). During training, performance metrics were monitored, enabling dynamic adjustment of hyperparameters to optimize results.

The validation phase involved evaluating the trained model on the validation set, analyzing metrics such as mAP, precision, and recall. This ongoing evaluation helped identify potential improvements and ensured the model maintained consistent performance on unseen data, guaranteeing its generalization capability.

3.4 Model Inference

After training, the model was applied to 417 images captured using a GoPro Fusion camera with a fisheye lens, in the surroundings of the Hansahoehe building. To combine the outputs from the custom model (sky detection) and the COCO model (mobile objects), a Python script was developed to run inference with a confidence threshold set to 0.2. This value was selected to balance the model's sensitivity, ensuring effective detection of undesirable elements without introducing excessive false positives.

The script generates binary masks in image format (JPG or PNG), in which areas useful for photogrammetry, such as buildings and ground, are represented in white (value 255), and unwanted regions—including sky, people, and vehicles—are marked in black (value 0).

3.5 Integration into the Photogrammetry Pipeline

To evaluate the impact of removing unwanted regions from the photogrammetric process, 417 images of the Hansahoehe building were used, each paired with its corresponding binary mask. These images and masks were processed using Agisoft Metashape Professional Edition, which allows importing masks associated with input images to define regions to be ignored during processing.

In Metashape, each image was linked to its respective mask file, ensuring that elements such as sky, people, and vehicles were correctly identified and excluded from subsequent steps. This association directs the software to consider only regions of interest in the images, enhancing 3D reconstruction accuracy.

Additionally, four Ground Control Points (GCPs) with known coordinates and 25 Check Points (CPs) were included. GCPs were used to georeference and scale the model during the photogrammetric process, while CPs served to assess the accuracy of the generated products.

Model accuracy was evaluated by calculating the Total Error of both GCPs and CPs, expressed in centimeters. The GCP total error reflects the average discrepancy between the known control point coordinates and the coordinates estimated by the model. Similarly, the CP total error indicates the average difference between the actual and estimated check point coordinates. These errors were computed using the Root Mean Square Error (RMSE) formula, Equation (12), where n is the number of points and d_i is the difference between the measured and estimated coordinates for point i (Queiroz & Gomes, 2001).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(d_i)^2} \quad (12)$$

During photo alignment, the Structure from Motion (SfM) algorithm detects keypoints in each image and matches them across different photos to identify tie points. The masks ensure that only unmasked areas contribute to this matching process, preventing unwanted elements from influencing the alignment.

The quality of alignment is quantified by the Root Mean Square Reprojection Error (Reprojection RMS), measured in pixels. This value represents the average discrepancy between the reprojected positions of tie points and their observed positions in the images. A lower RMS indicates more accurate alignment (HARTLEY & ZISSERMAN, 2004).

After alignment, the software generates a dense point cloud representing the surface of the object of interest. The density of this point cloud, expressed in points per square meter (pts/m²), depends on the image resolution and overlap. A higher point density results in a more detailed representation of the surveyed object.

During processing, outlier points were identified and removed to improve model quality. Outliers are points that deviate significantly from the general data pattern and may result from measurement errors or incorrect matches. Removing these points is essential to reduce noise and artifacts in the final model.

The use of masks in the photogrammetry workflow aims to improve the quality of 3D reconstruction by minimizing the interference of undesired elements and optimizing the modeling process. The results of this approach are detailed in Chapter 5.

4. Results

This chapter presents the results obtained from applying the YOLOv8 model for the automatic detection of undesirable elements in terrestrial photogrammetry images. The results are organized into three main sections: (1) performance of the YOLOv8 model during training and validation, (2) generation of binary masks from new images, and (3) impact of using masks in the digital photogrammetry pipeline. The evaluation metrics include **precision**, **recall**, **mAP@50**, **reprojection RMS error**, **point cloud density**, and **total processing time**.

4.1 YOLOv8 Training for Sky Detection

During the training of the YOLOv8n-seg model for automatic segmentation of sky areas in terrestrial images, data was collected over 50 epochs, with real-time monitoring of metrics and a final evaluation using the optimized model. By the 50th epoch, the model achieved the results shown in Table 1.

Table 1 – YOLOv8 training metrics at the 50th epoch.

Metric	Box (Detection)	Mask (Segmentation)
Precision (P)	92,4%	92,3%
Recall (R)	53,3%	53,3%
mAP@50	71,8%	67,8%
mAP@50-95	52,4%	52,1%

Source: Authors (2025).

The high precision (approximately 92%) indicates that the model rarely misclassifies non-sky regions as sky, minimizing false positives. However, the recall of 53.3% suggests that the model may fail to detect all sky areas, especially in complex scenarios where the sky is partially obstructed by vegetation or structures. The mAP@50 of 67.8% for segmentation reflects an average 50% overlap between predicted and ground truth masks, which is adequate for photogrammetric applications that require precise exclusion of non-structural regions. The results obtained were compared with similar studies that used YOLOv8 for segmentation tasks, as shown in Table 2.

Table 2 – Comparison with similar studies using YOLOv8.

Study	mAP@50 (Segmentation)	Application
This work	67,8%	Sky in terrestrial images
Wang et al. (2023)	72,1%	Urban objects in remotely piloted aircraft imagery
Zhang et al. (2023)	65,4%	Vegetation segmentation

Source: Authors (2025).

The model's performance in this study is consistent with the existing literature, indicating reliable results. However, there is room for improvement, particularly in low-contrast scenarios, such as overcast days, where distinguishing the sky from other elements becomes more challenging.

4.2 Binary Mask Generation

The average inference time for sky detection was 9.5 milliseconds, and for other elements (people, animals, and vehicles), it was 12 milliseconds. These values suggest the potential for implementation in real-time systems. However, for each specific case, testing on the available hardware would be necessary to confirm performance.

Figure 4 – Original image on the left and its overlay with masks on the right.
Source: Authors (2025).

By visually analyzing the masks overlaid on the original images, it is evident that the method is effective for most of the sky area. Some edges slightly intrude into other objects (such as trees or buildings), while in other areas they fall short. Figure 4 presents one such case, where it can be observed that overhead cables are ignored. There are also instances where elements like walls and roofs, with textures and colors very similar to the sky, were misclassified as sky (Figure 5).

Figure 5 – Overlay of masks on original images.
Source: Authors (2025).

4.3 Impact on the Photogrammetry Pipeline

The introduction of masks generated by the YOLOv8 model into the photogrammetric reconstruction workflow in Agisoft Metashape had contrasting effects on the model's geometric quality and computational efficiency. Table 3 summarizes the main metrics obtained with and without the use of masks.

Despite improvements in aspects such as reprojection error, point cloud density, and processing time, an increase in positional errors of Check Points (CPs) was observed, suggesting a disturbance in the geometric balance of the photogrammetric block.

The Reprojection RMS—which measures the average deviation between the observed and reprojected positions of tie points—was reduced from 0.944 to 0.897 pixels, representing a 5.2% improvement. This reduction indicates that the model became more internally consistent, as the masks prevented non-informative regions (such as sky or moving objects) from interfering with the matching of corresponding points between images.

The point cloud density also improved, increasing from 496.7 pts/m² to 521.5 pts/m²—a 5.0% gain. This suggests that by eliminating irrelevant areas, the algorithm was able to focus processing on textured and structurally meaningful regions. Additionally, the total number of points increased by 3.2%, and the proportion of removed outliers decreased from 15.7% to 12.3% (a 21.7% reduction), indicating that the model generated with masks was cleaner and less noisy.

The total processing time was also reduced by 6.0%, going from 21.8 minutes to 20.5 minutes, highlighting superior computational efficiency due to the avoidance of irrelevant regions during processing.

However, the positional accuracy metrics revealed an opposite trend. The total error for Check Points (CPs) increased from 8.25 cm to 9.05 cm, a deterioration of 8.8%. Likewise, the error for Ground Control Points (GCPs) went from 1.18 cm to 1.19 cm, although the 0.8% difference is marginal. The decline in external geometric performance can be attributed to the accidental exclusion of important building areas necessary for the stabilization of the photogrammetric block—especially upper facade sections with light colors or glazed areas, which were incorrectly classified as sky by the model.

This exclusion compromised the quantity and distribution of tie points in these critical regions, weakening the block structure and negatively impacting spatial triangulation. As a result, the model reconstructed with masks, while cleaner and more internally coherent, showed lower global positional accuracy.

Table 3 – Comparative metrics between photogrammetric processing results.

Metrics	With Masks	Without Masks	Improvement
GCPs Total Error (cm)	1,19	1,18	-0,8%
Check Points Total Error (cm)	9,05	8,25	-8,8%

Metrics	With Masks	Without Masks	Improvement
RMS (pix)	0,897	0,944	5,2%
Total points	902.266	874.146	3,2%
Densidade (pts/m²)	521,5	496,7	5,0%
Outliers Points Removed	12,3%	15,7%	21,7%
Total Time (min)	20,5	21,8	6,0%

Source: Authors (2025).

5. Discussion

The experiments conducted demonstrate that automatic segmentation using YOLOv8 can enhance internal aspects of the photogrammetry pipeline, while also introducing challenges in positional accuracy control. The application of masks resulted in a significant improvement in the internal quality of the model, with a 5.2% reduction in reprojection error, a 5.0% increase in point cloud density, a 3.2% increase in the total number of reconstructed points, and a 21.7% reduction in the proportion of outliers. These indicators suggest that removing regions such as sky, vehicles, and people contributed to a cleaner and more efficient 3D model, with less interference from undesirable elements.

Additionally, a reduction in total processing time was observed—from 21.8 minutes to 20.5 minutes (a 6.0% decrease). This performance gain is linked to the prior exclusion of non-informative areas, allowing the software to focus its processing resources on surfaces relevant to reconstruction.

However, despite these internal improvements, external accuracy metrics showed the opposite trend. The total error in check points (CPs) increased from 8.25 cm to 9.05 cm (an 8.8% deterioration), and the error in ground control points (GCPs) rose slightly from 1.18 cm to 1.19 cm (0.8%). This discrepancy indicates that, although the model became more internally coherent, there was a compromise in the geometric rigidity of the photogrammetric block relative to the external reference system.

A possible explanation for this decline in accuracy is that in some images, important facade areas of the building were mistakenly classified as sky and removed during segmentation. This exclusion led to a reduced number of keypoints in these structural regions, affecting the triangulation and spatial locking of the block. The result is a slight degradation in positional accuracy, especially in vertical areas with repetitive geometry or low texture.

It is important to highlight that this loss of accuracy could be mitigated by increasing the number of GCPs inserted into the model. Having more well-distributed GCPs would help stabilize the photogrammetric block and refine the external orientation, compensating for any loss in automatic triangulation. However, this strategy demands more fieldwork, significantly increasing the time required for data collection, georeferencing, and marking. In new surveys—especially those with operational constraints or covering large areas—this solution becomes less feasible, which is why improving automatic segmentation and selectively reviewing masks remain more efficient and scalable alternatives.

These results underscore the importance of carefully calibrating the segmentation model to avoid the exclusion of critical surfaces. In applications that require high geometric precision, manual mask review or the use of specialized classifiers for urban architecture may be necessary to mitigate the adverse effects of automated segmentation.

6. Conclusion

This study validated an automated methodology for removing unwanted elements in terrestrial images by integrating semantic segmentation with YOLOv8 into the Structure-from-Motion (SfM) process. The trained model achieved satisfactory performance in sky segmentation (*mAP@50 = 67.8%*), reducing the number of outliers in the point cloud by 21.7% and the reprojection RMS error by 5.2%. A 6.0% reduction in total processing time was also recorded, along with a 5.0% increase in point density and a 3.2% increase in total reconstructed points. These results confirm that the proposed automation is effective in dynamic urban scenarios, improving the structural quality of point clouds and optimizing computational performance.

However, the analyses also showed that the automatic application of masks can compromise the positional accuracy of the final products. An 8.8% increase in check point error was observed, attributed to the unintended exclusion of structural building areas mistakenly classified as sky. This removal reduced the available information for alignment and spatial locking of the photogrammetric block, impacting the geometric stability of the solution.

Although one possible strategy to mitigate this impact is to increase the number of GCPs, this approach requires more time and effort for field data collection, which may render it unfeasible in new surveys—especially in contexts with operational limitations. Thus, it is crucial to calibrate the segmentation model to preserve surfaces critical to reconstruction, and to implement validation mechanisms or assisted mask review procedures.

For future work, it is recommended to explore hybrid architectures (e.g., YOLOv8 combined with Swin Transformers) that can offer more refined segmentation in urban contexts. It is also essential to expand the training datasets, including images captured under different atmospheric conditions and with texture and lighting variations, to improve the model's robustness and generalization capacity. Integration with emerging tools such as Segment Anything (KIRILLOV et al., 2023) may provide interactive mask refinement and allow for more context-aware segmentation.

In summary, the proposed approach eliminates manual image preparation steps, improves the structural quality of 3D models, and aligns with contemporary demands for intelligent, efficient, and replicable workflows—particularly in the documentation of urban environments and architectural heritage.

References

AGÜERA-VEGA, F.; CARVAJAL-RAMÍREZ, F.; MARTÍNEZ-CARRICONDO, P. Assessment of photogrammetric mapping accuracy based on variation ground control points number using unmanned aerial vehicle. *Measurement: Journal of the International Measurement Confederation*, v. 98, 221-227, 2017.

BOESCH, G. YOLOv8: A Complete Guide [2025 Update]. *In Viso.ai*. Disponível em: https://viso.ai/deep-learning/yolov8-guide/. Acesso em: 02/03/2025.

Brostow, G. J.; Fauqueur, J.; Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, v. 30, n. 2, 88-97, 2009.

Goodfellow, I.; Bengio, Y.; Courville, A. *Deep learning*. Cambridge: MIT Press, 2016. 775p.

Grussenmeyer, P.; Alby, E.; Landes, T.; Koehl, M.; Guillemin, S.; Hullo, J. F.; Assali, P.; Smigiel, E. Recording Approach of Heritage Sites Based on Merging Point Clouds From High Resolution Photogrammetry and Terrestrial Laser Scanning. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, v. XXXIX-B5, 553-558, 2012.

Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*. 2. ed. Cambridge: Cambridge University Press, 2004. 672p.

JUNCKLAUS MARTINS, B.; POLLI, M.; CERENTINI, A.; MANTELLI, S.; CHAVES, T.; MOREIRA BRANCO, N.; VON WANGENHEIM, A.; ARRAIS, J. *Clouds-1000*. *In Mendeley Data*. Disponível em: https://data.mendeley.com/datasets/4pw8vfsnpx/1. Acesso em: 16/02/2025.

Kateb, F. A.; Monowar, M. M.; Hamid, M. A.; Ohi, A. Q.; Mridha, M. F. FruitDet: Attentive feature aggregation for real-time fruit detection in orchards. *Agronomy*, v. 11, n. 12, 1-21, 2021.

KINGMA, D. P.; BA, J. L. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1-15, 2015.

KIRILLOV, A.; MINTUN, E.; RAVI, N.; MAO, H.; ROLLAND, C.; GUSTAFSON, L.; XIAO, T.; WHITEHEAD, S.; BERG, A. C.; LO, W. Y.; DOLLÁR, P.; GIRSHICK, R. Segment Anything. *Proceedings of the IEEE International Conference on Computer Vision*, 3992-4003, 2023.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft COCO: Common Objects in Context. *Computer Vision - ECCV 2014: 13th European Conference*, Zurich, Switzerland, 740-755, 2014.

Macuácua, J. C.; Centeno, J. A. S.; Firmino, F. A. B.; Crato, J. K. T. Do; Vestena, K. de M.; Amisse, C. Automatic detection of urban infrastructure elements from terrestrial images using deep learning. *Boletim de Ciências Geodésicas*, v. 30, 2024.

MILLETARI, F.; NAVAB, N.; AHMADI, S. A. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016*, 565-571, 2016.

Musleh, A.; Alryalat, S. A.; Qasem, A. Image Annotation Software for Artificial Intelligence Applications. *High Yield Medical Reviews*, v. 1, n. 2, 1-5, 2023.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, v. 2016-Decem, 779-788, 2016.

Remondino, F.; Campana, S. *3D Recording and Modelling in Archaeology and Cultural Heritage:* Theory and best practices. Ann Arbor: University of Michigan Press, 2020. 352p.

Snavely, N.; Seitz, S. M.; Szeliski, R. Modeling the world from Internet photo collections. *International Journal of Computer Vision*, v. 80, n. 2, 189-210, 2008.

Son, H.; Kim, C. 3D structural component recognition and modeling method using color and 3D data for construction progress monitoring. *Automation in Construction*, v. 19, n. 7, 844-854, 2010.

ULTRALYTICS. YOLOv8 - Ultralytics YOLO Docs. *In Ultralytics Documentation*. Disponível em: https://docs.ultralytics.com/models/yolov8/. Acesso em: 16/02/2025.

WANG, C. Y.; MARK LIAO, H. Y.; WU, Y. H.; CHEN, P. Y.; HSIEH, J. W.; YEH, I. H. CSPNet: A new backbone that can enhance learning capability of CNN. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, v. 2020-June, 1571-1580, 2020.

Westoby, M. J.; Brasington, J.; Glasser, N. F.; Hambrey, M. J.; Reynolds, J. M. "Structure-from-Motion" photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, v. 179, 300-314, 2012.

Yunpeng, G.; Rui, Z.; Mingxu, Y.; Sabah, F. YOLOv8-TDD: An Optimized YOLOv8 Algorithm for Targeted Defect Detection in Printed Circuit Boards. *Journal of Electronic Testing: Theory and Applications (JETTA)*, v. 40, n. 5, 645-656, 2024.

ZEMKE, M. Símbolo histórico de Ibirama, edifício Hansahoehe completa 88 anos. *In Portal Educadora*. Disponível em: https://www.portaleducadora.com/noticia/simbolo-historico-de-ibirama-edificio-hansahoehe-completa-88-anos/. Acesso em: 02/03/2025.

ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. MixUp: Beyond empirical risk minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 1-13, 2018.

ZHENG, Z.; WANG, P.; LIU, W.; LI, J.; YE, R.; REN, D. Distance-IoU loss: Faster and better learning for bounding box regression. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, n. 2, 12993-13000, 2020.