

Scale-Invariant Object Contour Points (SIOCP): Integrating Instance Segmentation, Computer Vision and Geometric Refinement for Exact Contour and Keypoint Extraction of Façade Elements from Urban Images

Florian Frank¹, Venus Shah¹, Ludwig Hoegner², Rico Richter³

¹ IWT Friedrichshafen, Department of Mobility, Germany - frank@iwt-bodensee.de; shah@iwt-bodensee.de

² Hochschule München University of Applied Sciences, Department of Geoinformatics, Germany - ludwig.hoegner@hm.edu

³ University of Potsdam, Digital Engineering Faculty, Potsdam, Germany - rico.richter.1@uni-potsdam.de

Keywords: Invariant Feature Extraction, Image Instance Segmentation, 2D RGB Image Processing, 6DoF Pose Estimation, LoD3

Abstract

Exact 6DoF pose estimation is key to achieving high-quality CityGML LoD3 and BIM façade element reconstruction from urban monocular 2D RGB images. We present an implementation and experimental evaluation of our extraction pipeline for Scale-Invariant Object Contour Points (SIOCP). Our pipeline fuses processed information from 2D RGB images, IMU, GNSS+RTK, and LoD2 data to derive precise contours of regularly shaped objects and stable keypoints for 6DoF pose estimation. The image-processing pipeline is implemented using YOLOv8, SAM, and enhanced classical algorithms. Furthermore, the keypoint descriptor is based on a hierarchical object catalog that fully describes façades and the interdependencies of their elements in spatial and temporal contexts. This descriptor methodology enables reliable keypoint matching across texture-rich and detailed façade images from different perspectives, where classical methods partially fail in the benchmark. The implementation is still being refined. In summary, we present SIOCP as a key component for accurate 6DoF pose reconstruction, with future integration planned for LoD3 and beyond in building reconstruction.

1. Introduction

Image processing and edge AI have reached the mass market. This cross-sectional technology enables greater independence for users and can be considered a catalyst for new applications. Example applications include barrier-free autonomous navigation, do-it-yourself accessible Level of Detail 3 (LoD3) (Gröger et al., 2012) building model reconstruction for energy-efficient renovation, urban planning, or simulation. However, LoD3 building reconstruction has not yet reached the mass market. This is mainly due to the lack of cost-effective real-world digitization technologies, which are mainly based on mobile laser scanning (MLS) and semi-automated data processing. As a result, only a limited number of real-world European LoD3 datasets are available, such as SAVE (Schwab B. and Wysocki O., 2021) and TUM2TWIN (Wysocki et al., 2025). Addressing the challenge of cost efficiency has led to photogrammetric approaches and data migration techniques for generating LoD3 building façades from panoramic street-view images (Ma et al., 2026, Tang et al., 2025). A key prerequisite for this is accurate 6DoF pose estimation due to keypoint matching. The problem we address is that most existing feature-level keypoint extractors do not extract sufficient points for matching 6DoF-varied images of the same building façade.

To address this challenge, we propose a novel implemented pipeline for the extraction of scale-invariant object contour points from monocular 2D RGB images. The method is based on façade element instance segmentation, image processing, LoD2, and IMU data. This paper contributes a key component intended to simplify future building reconstruction at LoD3 and beyond. In particular, it seeks to lower the technological entry barrier by enabling implementation on handheld devices such as smartphones. Furthermore, the proposed approach supports future object-based applications in environmental perception and interpretation.

2. Related Work

This work originated from the idea of autonomous object-based outdoor localization by a mono-camera system. However, this idea initially faced a 'chicken-and-egg problem', because neither object-based datasets nor relocalization algorithms were available. Consequently, a handcart equipped with a 720° gimbal, a monocular camera, and an IMU was built for precise environmental image acquisition. Additionally, integrated GNSS RTK enables an absolute 6DoF pose estimation (Frank et al., 2024). Using the handcart, the façades of the Fallenbrunnen campus in Friedrichshafen and the TUM Roggenstein experimental farm were digitized using 12 MP images taken at 5° intervals. The high image resolution enables detailed façade element instance segmentation into 39 classes (Frank et al., 2024a). Based on this annotated dataset, a methodological reconstruction pipeline for LoD3 and BIM façade modeling was invented (Frank et al., 2025). A key component of the pipeline is the extraction of scale-invariant object-contour points for accurate 6-DOF pose estimation for each image.

This is where our paper contributes by implementing the SIOCP pipeline. Panoramic street view image approaches for LoD3 façade opening reconstruction (Tang et al., 2025, Ma et al., 2026) utilize spatial geometric information from 3D building models for the orthogonal reprojection of rectified image tiles. These orthogonal images are segmented using Semantic SAM or TeraNet. The Texture2LoD3 model is inspired by Zhu et al. (Zhu et al., 2022) for the best way of image projection. This enables tile stitching using SIFT and RANSAC, as well as ray casting of the quadrilateral fitted mask. In contrast, Ma et al. perform manual cropping to mitigate distortions, privacy masks, and blurriness for the best matching. Subsequently, camera parameters are estimated to enable the spatial reconstruction of façade elements.

Both approaches differ from the SIOCP embedded reconstruction approach (Frank et al., 2025), which directly estimates façade elements within the high-resolution 12 MP images and reconstructs the building utilizing their georeference and Structure from Motion (SfM). The success of image stitching and SfM fundamentally relies on robust feature point extraction and matching. However, when evaluating well-established feature extraction algorithms—such as ORB (Rublee et al., 2011), SIFT (Lowe, 2004), FAST (Rosten and Drummond, 2006), FREAK (Alahi et al., 2012), BRIEF (Calonder et al., 2010), and BRISK (Leutenegger et al., 2011)—alongside common matching techniques (NNS (Arya et al., 1994), KNN (Cover and Hart, 1967), FLANN (Muja and Lowe, 2009), and RANSAC (Fischler and Bolles, 1981)), significant limitations became apparent. Applying these methods to the Fallenbrunnen and Roggenstein images yielded poor results. This was due to the many repetitive patterns in the images, which led to too many features being extracted, making it difficult for RANSAC to find reliable matches. Finally, a 6DoF pose estimation starting with eight points (Longuet-Higgins, 1981, Hartley, 1997) does not produce acceptable results.

To address these challenges, SIOCP was invented to systematically extract and match keypoints in façade images. The SIOCP is a byproduct of 2D image photogrammetry building reconstruction utilizing classical computer image processing algorithms from OpenCV (Bradski, 2000), Scikit (van der Walt et al., 2014). Additionally, it leverages deep learning, such as YOLOv8-World (Cheng et al., 2024) for object detection and SAM (Kirillov et al., 2023) for instance segmentation.

3. Methodology

This section outlines a SIOCP-based methodology for practical implementation. The first subsection provides a formal definition of the SIOCP, followed by a strategic subsection describing the SIOCP extraction workflow. Finally, we present an updated system architecture, which leverages enhanced modules to optimize feature extraction from urban street-view images.

3.1 Definition of SIOCP

The definition of a SIOCP is a unique and stable point in both a 2D RGB image and 3D space. The single SIOCP is linked to an object and its relative position on a planar surface. The matching of SIOCPs is performed using inter-object dependencies in 2D images. In particular, we propose that SIOCPs typically correspond to rectilinear object contour edge points. These edges are typically derived from structures such as buildings, windows, doors, casements, and balconies. Consequently, a single straight line, defined by its two endpoints, results in two SIOCPs.

3.2 Strategic Description of the SIOCP Extraction Pipeline

This subsection introduces the general image processing workflow required for SIOCP extraction to provide a clearer understanding of the methodology. The simplified sequence of these individual steps is visualized in Figure 1. Before describing the SIOCP extraction steps, the handcart-based data collection system is introduced. The data collection system consists of a handcart-mounted 360° gimbal with a 12 MP Sony IMX477 camera and 16 mm lens. Images are acquired at 5° intervals,

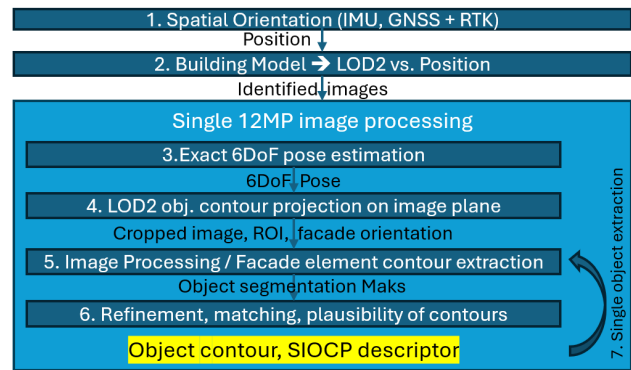


Figure 1. Block diagram of the SIOCP extraction pipeline.

yielding georeferenced high-resolution images with associated 6DoF poses (Frank et al., 2024).

Starting with the first step, the general millimeter-exact position and spatial orientation of the handcart and its gimbal-mounted camera is estimated by GNSS-RTK and IMU data. The spatial ego-position is used in the second step to identify, from the point of view of the handcart, the directly visible surrounding CityGML LoD2 buildings along the line of sight. Each building is used to identify by its outer contour in azimuth and elevation to relate images from the point of view of the handcart. The next step begins with the single image processing for SIOCP extraction. Beginning with the third step, the exact single-image 6DoF pose is estimated. The 6DoF pose is used in the fourth step to crop the image ROI according to the LoD2 building-related wall due to image plane projection. Further, the VP of the wall is determined for filtering. The fifth step is the image processing façade object instance segmentation and its contour extraction. Step six is the fusion of the object instance mask, its contour from the image, and LoD2 information. The matched contour is afterwards refined on the pixel sublevel to improve accuracy. Further, the plausibility of inter-object dependencies (Frank et al., 2024a) is checked. This also leads to the SIOCP, which references the point by one or more lines, buildings, façades, and hierarchical object relations. The seventh step indicates the iterative procedure of hierarchical object extraction of the façade elements. An example of inter-object dependency is a building wall, a window casement, a window frame, and a glass.

3.3 System Architecture Adaptation

The overall system architecture of the proposed SIOCP extraction is illustrated in Figure 2. The purpose of this section is to provide an overview of the various enhanced modules and processing domains for SIOCP extraction for practical implementation. Thus, modifications and extensions of the original concept (Frank et al., 2025) are described. Detailed descriptions of the individual modules are provided in the implementation section 4. The architecture is divided into four domains. Three of the domains rely on prior main information sources and their processing sensor data, building a model and an image processing. The fusion domain integrates outputs from the sensor, image processing, and building-model domains to generate SIOCPs, edge lines, and additional information for the subsequent 3D reconstruction process. The sensor processing domain is enhanced by a coordinate transformation module. The gimbal module is also able to deliver a horizon and 6DoF with every image. Further, an excentric drift compensation was

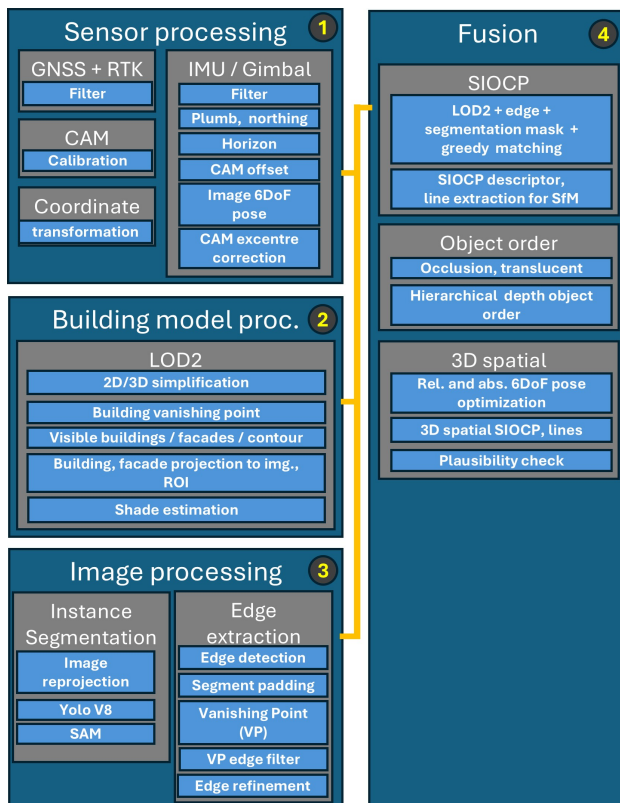


Figure 2. System architecture of single components and their dependencies.

invented to minimize orientation and northing errors in multiple images. The building model domain was enhanced with a 2D and 3D simplification for visible building and façade estimation from a point of view. Building vanishing point extraction enables shade estimation and is used in the image processing domain for edge filtering or image reprojection, for instance segmentation. The image processing domain was updated across all modules. In addition, an image VP edge extractor was implemented for buildings without LoD2 information. The fusion domain is a completely new implementation of all components. Detailed information is provided in the next section.

4. Implementation and Integration SIOCP

The implementation section details the practical implementation of single modules. Considering the various modules of the system architecture, this section focuses exclusively on the modules developed for SIOCP extraction.

4.1 Coordinate Systems

The extraction of SIOCPs depends on seven distinct oriented coordinate systems. The global reference is the World Coordinate System (WCS), which contains the handcart and the LoD2 building models. The handcart itself utilizes a vehicle coordinate system oriented according to DIN ISO 8855, serving as a reference for relative sensors. In addition, the GNSS+RTK system has a dynamic offset relative to the gimbal-mounted camera and IMU. Both sensors are eccentrically mounted on the gimbal. Therefore, a 6DoF pose correction for the camera was implemented. Finally, the image sensor plane coordinate system

has to be mentioned. To validate the system and its assumptions for SIOCP extraction, all coordinates are transformed into a local Cartesian ENU-oriented coordinate system.

The eccentricity correction compensates for the longitudinal and lateral rotational displacements of the camera between consecutive images.

The algorithm requires at least one common SIOCP in the two images. The two focal points form a baseline with known angles by the image plane and the camera matrix, and the gimbal rotation. The SIOCP acts as the third vertex, completing the triangle. The distance to the SIOCP is estimated by a projection onto the corresponding LoD2 building wall. Finally, the error is proportional to the depth estimation and its accuracy. For short distances of 9 to 10 m, a baseline of 4 cm by 1 m distance error leads to an angular error of approximately 46''.

4.2 LoD2 Feature Derivation

In Germany, modern city models are managed by municipal administrations and are mainly based on aerial photogrammetry, airborne laser scanning (ALS), 2D building footprints for cadastral records, and architect-created LoD2.3 models. Reusing this LoD2 data to create LoD3 and beyond façades is not a new idea (Frank et al., 2025, Tang et al., 2025, Ma et al., 2026). The outer building contour is derived through 2D and 3D simplification of the models, employing co-linearity and co-planarity checks. In this process, edges connecting two or more planar surfaces are identified for subsequent SIOCP extraction. Before 2D and 3D simplification, an achievable real-world accuracy threshold of 5 mm is defined by a coplanar pixel width of the image plane. Applied to the pinhole projection of a pixel IMX477 equipped with a 16 mm lens, a distance of around 51.65 m is calculated. The estimated distance is used to identify buildings and façades around the handcart ego-position.

Therefore, the LoD2 building models are projected orthogonally to the ground. The buildings and façades visible from the handcart's ego-pose within the Field of View (FoV) are first estimated using 2D projections in the polar coordinate system. In the second step, 3D raycasting is used to identify building façade-associated images of the 5° spherical gimbal records. By applying the inverse camera matrix, the outer contours, façades, and points of the LoD2 buildings are projected onto the image plane according to the 6DoF pose. This projection serves as a region of interest (ROI) in the image to improve performance and associate buildings, façades, and their elements with the image. In summary, this is the link between an image and a building model for SIOCP extraction. To refine the results, a coarse alignment of the LoD2 image plane projection is performed using the northing and horizon data provided by the sensor section.

The vanishing point module calculates LoD2 building façade-related vanishing points for an associated image and its 6DoF pose. The objective is to filter edges extracted from the image without requiring a quadrilateral orthogonal reprojection. A detailed description is given in the following image processing section.

4.3 Image Processing and Integration

The image-processing pipeline consists of object detection, instance segmentation, and CV algorithms.

The object detection and instance segmentation pipeline using YOLO-SAM follows the detailed idea and implementation from the research (Frank et al., 2025). In this research, the ground truth dataset was expanded to 570 images with object instances up to 15,104. Furthermore, hyperparameter tuning is used to fine-tune YOLOv8x-Worldv2 (Cheng et al., 2024) and YOLO11x (Jocher and Qiu, 2024), refining the bounding box prompts and improving detection accuracy.

Secondly, the CV algorithm modules, primarily based on OpenCV (Bradski, 2000) functions, were significantly refined to enhance overall accuracy. A new **edge detection** pipeline was implemented to improve the results. We tried a collection of different line detectors (Vincentqyw, 2023), however, the results never met our standards. We previously enhanced Canny edge detection using skeletonization and combined it with HoughLinesP (Frank et al., 2025).

The resulting vectors have different orientations, which is where our **padding algorithm** is applied. To reduce fragmented line detections, nearby vector segments with similar orientations are merged into a single representative segment:

$$l_i = (\mathbf{p}_{i,1}, \mathbf{p}_{i,2}), \quad \mathbf{p}_{i,k} = (x_{i,k}, y_{i,k})^T, \quad (1)$$

Its orientation is computed as,

$$\theta_i = \text{atan2}(y_{i,2} - y_{i,1}, x_{i,2} - x_{i,1}) \bmod \pi. \quad (2)$$

The modulo operation in (2) ensures that antiparallel directions are treated as equivalent. Two lines l_i and l_j are considered angularly compatible if their circular angle difference satisfies,

$$\Delta\theta_{ij} = \min(|\theta_i - \theta_j|, \pi - |\theta_i - \theta_j|) \leq \theta_{\max}. \quad (3)$$

In addition to the angular constraint in equation (3), spatial proximity is verified by testing whether at least one endpoint of either segment lies within a maximum perpendicular distance $d_{\max}$ to the other segment. Hence, two segments are grouped if they are both orientation-consistent and spatially adjacent. Starting from one segment, the group is expanded iteratively until no further compatible segments can be added.

For each final group, all segment endpoints are collected, and a single regression line is estimated by

$$y = mx + b, \quad (4)$$

using linear least squares over all points in the group. The merged segment is then defined by evaluating (4) at the minimum and maximum x -coordinates of the group. In this way, the method produces longer and more stable lines.

The perspective transformation is used to improve instance segmentation results by providing an orthogonal view of the planar walls of the façade. This process involves projecting the four quadrilateral points of an **LoD2** wall onto the image plane. Equation (5) projects a world point P_w to the image plane p . Therefore, P_w relative translation to the camera's central point C is calculated in the ENU coordinate system and transformed to the camera pose via its rotation matrix R . The camera matrix K is used to perform a perspective projection to the image plane.

$$p \sim K [R \mid -RC] \begin{bmatrix} P_w \\ 1 \end{bmatrix} \quad (5)$$

Lastly, the projected points on the image plane are used for building ROI and its edges and using quadrilateral points to warp the image for an orthogonal view (Bradski, 2000). Figure 4 illustrates this process in the second and third pictures.

The **line refinement** module is developed to improve the accuracy of the edge line at the sub-pixel level. The process begins with a grayscale conversion of the input image. A padded edge vector is overlaid, and along in equally distant steps, orthogonally the intensity level curve is analyzed. Figure 3 illustrates an edge with non-linear characteristics. At every equal distance step, five to eight orthogonal pixels are sampled on both sides of the edge. Along the orthogonal, a height of at least one pixel describes an area. For each pixel within this area, a new area-weighted intensity value is calculated from the grayscale source (black frame). These calculated values are then convolved with a Point Spread Function (PSF) to determine the exact inflection point of the edge intensity curve. In cases where multiple inflection points occur, the one exhibiting the highest gradient is selected. The PSF used in this model was empirically determined using a 1 mm aperture placed in front of an LED positioned at a distance of 20 m. Finally, a best-fit line is calculated

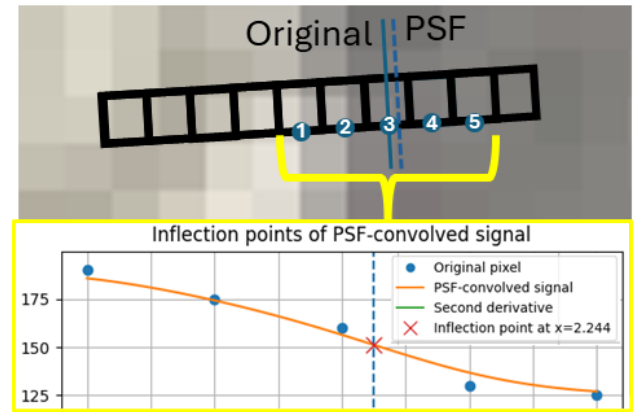


Figure 3. Example for PSF-based edge correction.

by PSF-determined inflection points and is assumed as an edge.

4.4 SIOCP Extraction and Uniqueness

Figure 4 illustrates the fusion process of the SIOCP extraction module, which combines results from instance segmentation, edge and line extraction, and LoD2 image plane projection. The extraction process for each identified building originates from its visible and longest outer vertical edges that intersect the ground plane. These edges serve as a reference to compensate for northing errors or vertical wall separations within the buildings. Around the LoD2 edges, an ROI is defined to find associated images with possible edges. This is necessary due to differences between the real-world building and LoD2 models. In a nutshell, the LoD2 model is used in SIOCP extraction just for ROI search determination and associating the SIOCP point itself with a building and a wall and may further sub-wall IDs from the LoD2 models.

The fusion starts with the projected wall of the LoD2 building to the image plane, where the ROI window is extended orthogonally to the edge. The matching in ROI is fulfilled by the following criteria, where two instance-segmented walls touch on an edge, and the edge endpoints must be unique in 3D space. A matching line is selected according to an iterative closest

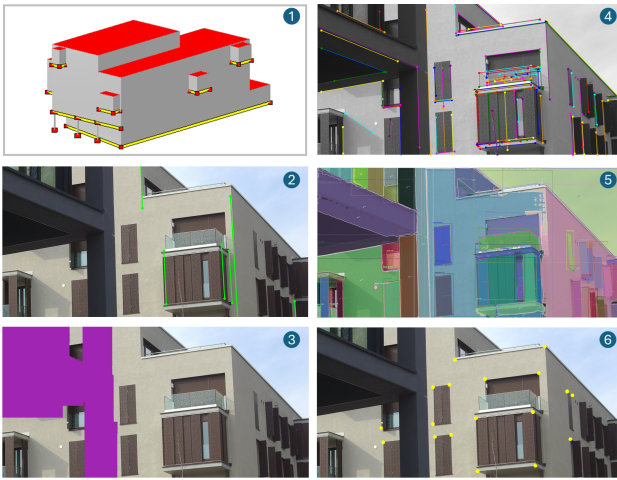


Figure 4. Enumerated, stacked images explaining the SIOCPs extraction process by order numbers: 1st LoD2 Building, 2nd vertical edges, 3rd LoD2 ROI mask, 4th extracted padded image lines, 5th image instance segmentation, 6th estimated SIOCPs

point (ICP) algorithm that brings together edge line vectors in length and position. Equation (6) formulates a nearest-neighbor matching under a distance constraint and a greedy one-to-one assignment.

$$m_j = \arg \min_{m \in M} \|(Rp_i + t) - m\|^2 \quad (6)$$

Figure 5 illustrates the **hierarchical linking of objects to the SIOCP**. The building (1) is linked to a wall (2). The wall has an edge and is linked to an edge, the next child is a window casement (3). Steps 2 to 3 are repeated iteratively for inheritance according to the object hierarchy for every new sub-object carrying new detailed information, e.g., a window frame. Finally, the last object links the SIOCP to an edge.

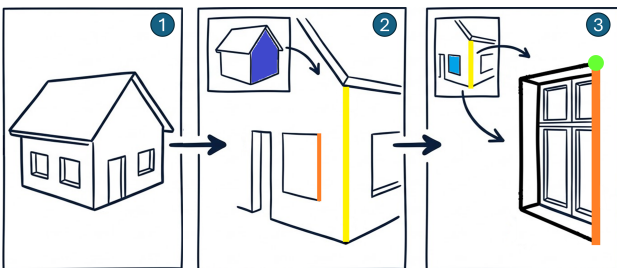


Figure 5. SIOCP hierarchical object class and edge linking: Building (1, root), wall + wall edge (2, child 1), window + edge to SIOCP (3, child 2)

The SIOCP extraction follows a top-down approach, from large (building) to detailed (window handle) objects. Depending on the object class, the edges are approached either internal or external at the ROI. Derived from LoD2 dimensions and vanishing points, the relative positions of sub-objects on the wall and further linked objects, as well as their IDs are stored in the SIOCP. Figure 6 illustrates the SIOCP descriptor, which contains information about object dependencies to SIOCP and referenced images. The inter-object descriptions establish a semantic link between the SIOCP of a primary object and its inherited sub-objects. Due to systematic extraction, there is no search algorithm, like RANSAC, necessary to connect multiple images.

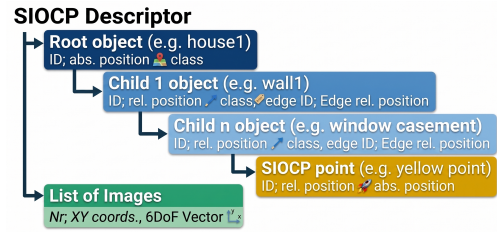


Figure 6. SIOCP descriptor referencing objects and images.

5. Experiments and Results

This section presents experiments and results specifically from modules and components developed for SIOCP extraction. Due to space constraints, this section focuses on the most significant results and key performance metrics.

The **coordinate system** registration is an essential prerequisite for 6DoF image processing. IMU and gimbal data are processed and combined with optical stabilization extracted from images. Table 1 summarizes the accuracy results from all involved components. The lateral position is in the sub-millimeter range. But the IMU drift offsets have to be compensated for by the plumb vector orthogonal plane horizon crossing the images. The horizon accuracy was determined by real-world captured images with horizon markings, marked with a leveling line laser. Furthermore, the inter-image rotational movement errors are the worst-case errors.

Datatype	Motion	Accuracy	Additional information
GNSS+RTK (fixed point)	Gimbal	translational ≤ 1 mm	by several 100 samples
Backlash compensated Gimbal	rotational	21"	
IMU (datasheet accuracy)	rotational	2°	
IMU (offset, drift in dataset)	rotational	3°	
PlumbVector	rotational	≤ 1 mm @ 1.2m plumb	by several 100 6DoF poses
Horizon (Plumb to Image 6DoF Pose)	rotational	1-3 pix	image review
Images to image (horizon or vertical)	rotational	2x1 pix = 39.22°	f = 16.38mm, pix _{width} = 1.55um
	rotational	≤ 46" @ 1 m depth error	eccentricity-induced worst-case error due to depth estimation

Table 1. Overview of sensor data and its accuracies.

LoD2 buildings are primarily used to estimate VPs, wall quadrilateral points, and ROIs for edge detection. The VP estimation shows a high reliability and delivers good results for line filtering. For validation, we projected building walls onto the image plane. Subsequently, the VPs of the buildings in the images were estimated by the intersection of the edge lines. Real-world images and LoD2 VPs are close together, their use in a VP-based line filter resulted in a maximum error of 0.83°. This result is applicable for general VP filtering and quadrilateral point image reprojection for instance segmentation. Applying LoD2-based ROIs for edge detection via image projection remains challenging because real-world buildings deviate from their LoD2 models in geometric dimensions and 6DoF poses. These discrepancies result in significant errors, for example, at Campus Fallenbrunnen, one building is nearly 5 m longer, and another one has a height offset of about 3 m. Figure 7 visualizes the height offset. Without correction or assumptions, ROI edge search is not generalizable. Therefore, most of the vertical LoD2 edges serve as features because they are mostly visible and can be matched in an image. The upper endpoints of these vertical edges are utilized to compensate for LoD2 height offsets, while the lower edges often end on uneven terrain.

In the **Image Processing – Object Detection and Instance Segmentation** module, 55 hyperparameter-tuning iterations were performed. The decreasing box losses on both the training and validation sets indicate improved object detection and

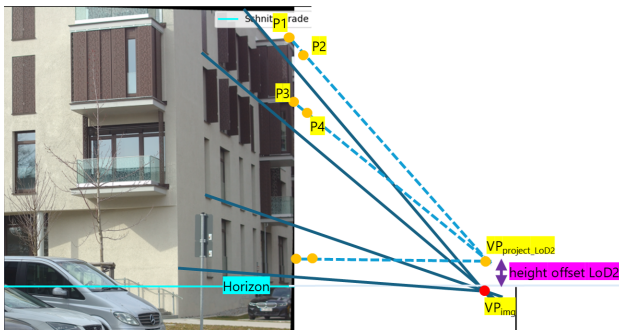


Figure 7. LoD2 offset in dataset projected on image, resulting in VP height offset

bounding box localization across 39 classes. However, the results of both models showed a slight overfitting toward the end of the 500 training epochs.

The distribution focal loss and classification loss converge slowly and remain relatively high for both the training and validation sets, likely due to class imbalance and the limited dataset size. Among the evaluated object detection models, YOLOv8x-Worldv2 outperforms YOLO11x. The best YOLOv8x-Worldv2 configuration was obtained at the 25th tuning iteration, achieving an mAP@50 of 0.768 and an mAP@50–95 of approximately 0.571. In comparison, the best YOLO11x configuration was obtained at the 44th iteration, with an mAP@50 of 0.715 and an mAP@50–95 of 0.487.

The **edge refinement** has been tested on a real-world-captured yield traffic sign. This traffic sign is a regular-shaped triangle, and our test case has a leg length of 90 cm. For the whole test case, ground truth data is available. The car was driving along on a road, and the triangle was in the FoV from 65 to 14 meters. At least the absolute distance of the triangle was estimated by normal and refined edges. The mean depth error of Canny edges decreases from 84.3 cm to 17.4 cm after refinement, which means a massive improvement in accuracy.

The **fusion module** has been experimentally evaluated in terms of fusion process performance and matchmaking. It runs on the same framework and computer system described in the previous SIOCP paper (Frank et al., 2025). In summary, AI image processing runs on average in 22 seconds; all edge detection, padding, refinement, LoD2 processing, and SIOCP data fusion take approximately 43 seconds on average.

The **SIOCP fusion process** was experimentally evaluated in a set of 20 images of Campus Fallenbrunnen. They have been selected by lighting shades, different object scales, and perspectives. The aim of the extraction process is to extract a stable SIOCPs from the images. The results of the processing step are shown in Table 2, and the maximum and minimum results of the SIOCP extraction are visualized in Figure 8.

Metrics per Image	Average	Minimum	Maximum
Amount of Objects	25.90	3	77
Object Confidence	0.7742	0.6791	0.9307
Padded Edges	413.1747	223	1003
Object matched edges	43.3236	4	109
Misaligned obj. edges	0.6327	0	2
Missing obj. edges	8.5796	0	33
Extracted SIOCP	34.7531	3	140

Table 2. Metrics per image of fusion process.



Figure 8. Images with minimum and maximum extracted SIOCP of the test data set of 20 images.

The instance segmentation module achieved an average object confidence of 0.6791 across the 39 distinct semantic classes. On average, the edge padding algorithm generated 413 edge candidates per image, of which 43 were successfully matched to the corresponding instance masks, and ~ 35 SIOCPs were extracted. The visual review of SIOCP and edges resulted in fewer than one misaligned object edge and fewer than 9 missing object edges per image.

The SIOCP extraction was invented to obtain absolute stable unique points in RGB 2D images for 3D 6DoF pose estimation and building reconstruction. The **comparison with other classical feature extractors and SIOCP** reveals that they have nothing in common in terms of their extraction processes and matching algorithms. The results in that classical benchmark are not applicable to our method (Bojanić et al., 2019, Isik, 2024). Finally, we tried to bring the different methods to a competitive level and apply them to a set of four images. Competitive, in this case, is that the different methods have to be challenged in finding stable keypoints for a 6DoF pose estimation. For this test, two images at the same position and two from different 6DoF poses with different perspectives and scales were selected. All images were captured from the same building façade and are not orthogonal. The best-performing feature extractors according to Bojanić et al (Bojanić et al., 2019) have been tested and their results are presented in table 3.

Method	Raw Matches	Good Matches	Inliers	Inlier Ratio
AKAZE	1835.36	136.71	39.29	0.31
FAST-SIFT	2337.96	187.07	43.46	0.20
ORB	1725.64	139.61	47.07	0.39
ROOTSIFT	2524.86	597.14	117.46	0.45
SIFT	2485.21	337.39	67.07	0.32
SIOCP	11.16	-	10.83	0.97

Table 3. Average matching performance over all evaluated image pairs.

The classical feature methods extracted several thousand feature points in each image. The RANSAC matching results in Table 3 show a significant discrepancy compared to the visually inspected inlier correspondence images. The texture-rich façades with rough plastering, darkened metal sheets with holes, and glass with vinyl points led to many confusions due to the pattern repeating. All classical algorithms just performed a useful correlation in the two images from the same position. In the other five image correlations, the classic algorithm underperformed in matching. The mathematically calculated inlier ratio is at least not meaningful, and a 6DoF pose estimation is not possible. Figure 9 shows the result of our methodically extracted SIOCP with a background of an edge-overlaid instance segmentation. Further, the images just show the most SIOCPs extracted by the longest edges. The representation of matching-line-correspondences of the four images is omitted to avoid confusion. Out of six image correlations, a total of 67 SIOCPs were



Figure 9. SIOCP matching points with instance-segmented façade elements and their matched edges in the background.

extracted. Only two SIOCPs were incorrectly classified as windows instead of as a frame class. A calculated inlier ratio of ~ 0.97 and the absence of false edge fusions demonstrate the precision of the proposed method. Finally, the reprojection error is not considered because the paper focuses on the SIOCP extraction and matching quality. The **object order** has not been evaluated in this paper.

6. Discussion

The SIOCP method was invented for 2D single-image to LoD3 and beyond building model reconstruction. Most buildings are **regularly shaped** with straight edges. Our method follows this assumption and currently supports only straight-edge lines. Other irregular line-shaped objects would require new or extended algorithms and are currently not supported, which limits the application of our method.

The **LoD2 building models** are integrated into the data processing for the spatial orientation of the façades and to enhance them to LoD3. But the models and real-world buildings differ significantly in appearance and dimension. In addition, not every real-world building has a digital LoD2 model, which limits the application of our method. A possible workaround would be to reconstruct the geometry of the outer building from VPs of images. The method of **PSF edge refinement** at the sub-pixel level is theoretically based on ideal, sharp, untextured edges, but this is an ideal assumption. Real-world images have irregular illuminations due to micro-shadows of plaster, edge bending radius, and asymmetrical changes in brightness caused by different viewing angles. Most edges are several hundred pixels long, allowing the determination of several orthogonal inflection points and the calculation of a regression line. In summary, PSF theory and practical observations differ. However, averaging enables the PSF-based refinement to function effectively for our application. Generally, AI-based **object detection** and **instance segmentation** require a lot of ground truth data. The instance-segmentation pipeline is currently optimized for the Campus Fallenbrunnen dataset. The TUM Rogenstein Farm was only partially considered due to its segmental arches. Ultimately, transferring the method to other object classes or datasets is not straightforward. New instance segmentation models may mitigate or compensate for this disadvantage. Furthermore, the nearly doubled processing time of 43 seconds for the CV implementation compared to AI-based object extraction is

caused by the prototype algorithm implementation in Python. In this paper, the modules of **occlusion** and **6DoF pose reconstruction** have not been evaluated. However, the referenced accuracy of the handcart system has been proven (Frank et al., 2024), and the padded lines for the determination of the accuracy have been visually and manually reviewed in the images. Only two lines were selected incorrectly in the **fusion process**, ending in 2 SIOCPs that shifted 2 pixels in the XY direction. The **fusion method** was evaluated by a set of images from the Campus Fallenbrunnen. The dataset consists of a mixture of different perspectives, scales, illuminations, 6DoF poses, and dates and times on the campus. Further, a test set of 20 images is not really representative according to the normal distribution of at least 35 images. However, the images were taken under optimized conditions of daytime and weather, without fog or rain. Accordingly, the robustness of the method under other environmental conditions remains unclear. The images were blurred by a 3x3 kernel before edge extraction to suppress micro-textures of the image and noise sensibility edge extraction. Compared to the other feature extractors that have been applied to the original image without optimization or adaptation, just with a standard configuration. The classical algorithms extracted several thousands of feature points from each image, but did not produce an applicable matching result in the four images. Relying on the confusing and repetitive pattern of façades and their elements. Finally, task-specific optimization might have led to better results.

In summary, we have shown that our approach of SIOCP extraction and matching works and can find relations in the images, where the classic feature extraction methods may fail.

7. Conclusion and Outlook

The paper presents the practical implementation and experimental results of our SIOCP approach, based on IMU, LoD2, GNSS+RTK data, and 2D RGB images. We improved the methodology and system architecture for SIOCP extraction by refining the existing processing steps and integrating additional components into a functional data processing pipeline. The main details about modules and their functionality are described in the implementation section. The fusion module merges the outputs of single modules. Resulting in a fully functional integration of the SIOCP extraction pipeline. Practical experimental evaluation showed that the implementation of the SIOCP method is effective in extracting keypoints from different images. Furthermore, the matchmaking of keypoints based on the SIOCP descriptor delivered proper results for the 8-point algorithm 6DoF pose reconstruction, where classic ones did not. We showed a possible alternative to classical and AI feature extractors (SIFT, ORB, AKAZE). This could be applied to ego-pose, object, or LoD3 and beyond reconstruction. Nevertheless, improving SIOCP is an ongoing task, as mentioned in the discussion section. We present a working and experimentally tested proof of concept, which seamlessly integrates into building reconstruction workflows for LoD3 and beyond. Furthermore, we plan to improve system accuracy using standardized referenced objects. Our long-term vision is to drive environmental perception for autonomous systems and urban modeling forward by simplifying these advanced digitization processes for use on standard handheld devices.

References

Alahi, A., Ortiz, R., Vandergheynst, P., 2012. Freak: Fast retina keypoint. *Proceedings of the IEEE Conference on Computer*

Vision and Pattern Recognition (CVPR), 510–517.

Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., Wu, A., 1994. An optimal algorithm for approximate nearest neighbor searching. *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '94, Society for Industrial and Applied Mathematics, USA, 573–582.

Bojanić, D., Bartol, K., Pribanić, T., Petković, T., Donoso, Y. D., Mas, J. S., 2019. On the comparison of classic and deep keypoint detector and descriptor methods. *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, 64–69.

Bradski, G., 2000. The OpenCV Library. *Dr. Dobbs's Journal of Software Tools*, 25(11).

Calonder, M., Lepetit, V., Strecha, C., Fua, P., 2010. Brief: Binary robust independent elementary features. *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, 778–792.

Cheng, T., Song, L., Ge, Y., Liu, W., Wang, X., Shan, Y., 2024. Yolo-world: Real-time open-vocabulary object detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16901–16911.

Cover, T., Hart, P., 1967. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1), 21–27.

Fischler, M. A., Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6), 381–395.

Frank, F., Hoegner, L., Buckel, P., Dalm, K., 2024. An alternative raw data acquisition approach for reconstruction of lod3 models. T. H. Kolbe, A. Donaubaue, C. Beil (eds), *Recent Advances in 3D Geoinformation Science*, Springer Nature Switzerland, Cham, 459–477.

Frank, F., Richter, R., Hoegner, L., 2024a. A concept for methodical classification, assignment of attributes, and labeling of facade elements in camera images for lod3 reconstruction of buildings. 44. *Wissenschaftlich-Technische Jahrestagung der DGPF*, Remagen, Germany, 181–190.

Frank, F., Shah, V., Worbis, S., Hoegner, L., 2025. 3D Façade Element Extraction from Image-based Instance Segmentation and Scale-Invariant Object Contour Points. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-G-2025, 473–480. <https://isprs-archives.copernicus.org/articles/XLVIII-G-2025/473/2025/>.

Gröger, G., Kolbe, T. H., Nagel, C., Häfele, K.-H., 2012. *OGC City Geography Markup Language (CityGML) Encoding Standard*. Open Geospatial Consortium.

Hartley, R., 1997. In Defense of the Eight-Point Algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6), 580–593.

Isik, M., 2024. Comprehensive empirical evaluation of feature extractors in Computer Vision. *PeerJ Computer Science*, 10, e2415. <https://pmc.ncbi.nlm.nih.gov/articles/PMC11623105/>.

Jocher, G., Qiu, J., 2024. Ultralytics yolo11. <https://github.com/ultralytics/ultralytics>. Software version 11.0.0, AGPL-3.0 license.

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., 2023. Segment anything. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4015–4026.

Leutenegger, S., Chli, M., Siegwart, R., 2011. Brisk: Binary robust invariant scalable keypoints. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2548–2555.

Longuet-Higgins, H. C., 1981. A Computer Algorithm for Reconstructing a Scene from Two Projections. *Nature*, 293, 133–135.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91–110.

Ma, R., Yang, C., Chen, J., Biljecki, F., Li, X., 2026. Street view imagery-based method for reconstructing 3D building façade openings. *Automation in Construction*, 184, 106842.

Muja, M., Lowe, D. G., 2009. Fast approximate nearest neighbors with automatic algorithm configuration. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 331–340.

Rosten, E., Drummond, T., 2006. Machine learning for high-speed corner detection. *European Conference on Computer Vision (ECCV)*, Springer, 430–443.

Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. Orb: An efficient alternative to sift or surf. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2564–2571.

Schwab B., Wysocki O., 2021. Lod3 road space models. <https://github.com/savenow/lod3-road-space-models>.

Tang, W., Li, W., Liang, X., Wysocki, O., Biljecki, F., Holst, C., Jutzi, B., 2025. Texture2LoD3: Enabling LoD3 Building Reconstruction With Panoramic Images. *arXiv preprint arXiv:2504.05249*.

van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Goullart, E., Yu, T., 2014. scikit-image: Image Processing in Python. *PeerJ*, 2, e453.

Vincentqyw, 2023. Linesegmentsdetection. <https://github.com/Vincentqyw/LineSegmentsDetection>. Accessed: 2026-03-17.

Wysocki, O., Schwab, B., Biswanath, M. K., Greza, M., Zhang, Q., Zhu, J., Froeh, T., Heeramaglore, M., Hijazi, I., Kanna, K., Pechinger, M., Chen, Z., Sun, Y., Segura, A. R., Xu, Z., AbdelGafar, O., Mehranfar, M., Yeshwanth, C., Liu, Y.-C., Yazdi, H., Wang, J., Auer, S., Anders, K., Bogenberger, K., Borrmann, A., Dai, A., Hoegner, L., Holst, C., Kolbe, T. H., Ludwig, F., Nießner, M., Petzold, F., Zhu, X. X., Jutzi, B., 2025. Tum2twin: Introducing the large-scale multimodal urban digital twin benchmark dataset. Version Number: 2.

Zhu, P., Para, W. R., Frühstück, A., Femiani, J., Wonka, P., 2022. Large-Scale Architectural Asset Extraction from Panoramic Imagery. *IEEE Transactions on Visualization and Computer Graphics*, 28(2), 1301–1316.