Deep Learning in Visual Odometry for Autonomous Driving

Luca Morelli[1], Paweł Trybała[1], Armando Vittorio Razzino[1,2], Fabio Remondino[1]

[1] 3D Optical Metrology (3DOM) unit, Bruno Kessler Foundation (FBK), Trento, Italy
Email: (lmorelli, ptrybala, arazzino, remondino)@fbk.eu

[2] Dept. Mathematics, Computer Science and Physics, University of Udine, Italy

Commission II

KEY WORDS: PNT, Visual-SLAM, Deep learning, Local features, Simulation, DROID-SLAM

ABSTRACT:

Positioning, Navigation, and Timing (PNT) solutions are fundamental for autonomous driving, ensuring reliable localization for safe vehicle control in diverse environments. While GNSS-based systems provide absolute positioning, they become unreliable in GNSS-denied scenarios such as urban canyons or tunnels. Dead reckoning techniques, including Visual Odometry (VO), offer an alternative by estimating motion from onboard sensors. Integrating these methods with deep learning (DL) has shown potential for enhancing robustness, particularly in challenging conditions. This study, part of the VAIPOSA ESA project, investigates the performance of VO solutions under various environmental conditions using a simulation-based approach. The CARLA simulator provides controlled testing scenarios, enabling the evaluation of VO accuracy across different weather conditions, illumination changes, and dynamic environments. A synthetic stereo setup enables capturing error-free ground truth trajectories and fair evaluation of the VO methods. Multiple sequences are analyzed, reflecting real-world challenges such as poor visibility, texture variations, and occlusions. The findings highlight the influence of environmental factors and dynamic objects on VO performance and the role of DL in mitigating common failure modes.

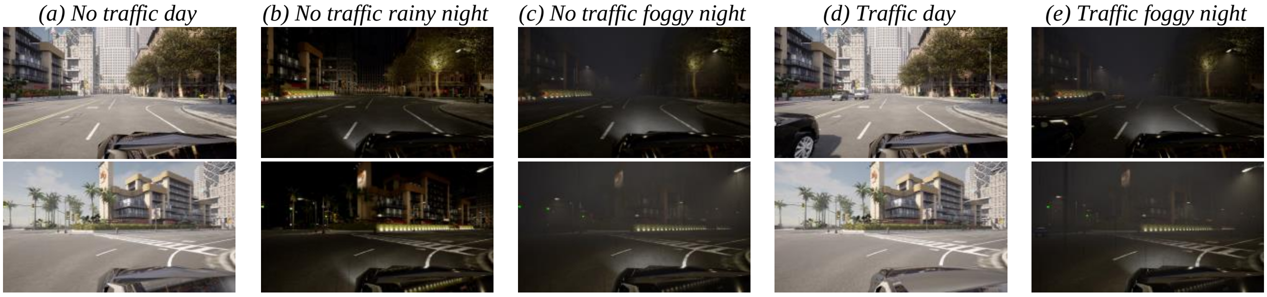


Figure 1: Samples of the five test scenarios with different light, weather and traffic conditions.

1. INTRODUCTION

1.1 Positioning and mobile mapping systems

Real-time positioning of a moving agent (i.e., determining its position, attitude, and their change in time) is a field of research of great interest with countless technical applications. It enables robotic platforms or vehicles to navigate through space, execute operations either semi-autonomously or fully autonomously, and collect spatially enriched data. Examples include precision agriculture (Weyler et al., 2023), autonomous vehicles (Yurtsever et al., 2020), and planetary exploration (Sanguino et al., 2017). A positioning system - carried by humans or animals - allows the study of movement patterns for security, medical, or commercial purposes (Correa et al., 2017; Aziz and Koo, 2025). Positioning systems typically integrate multiple sensors to enhance data redundancy and mitigate individual technological limitations (Fayyad et al., 2020). The cost and accuracy of these systems vary widely depending on the application. In open-sky environments, the preferred solution is the Global Navigation Satellite System (GNSS), which can provide accurate absolute positioning within a global reference frame. In the automotive sector, for instance, it is common to integrate GNSS with inertial measurement units (IMUs), LiDAR, RGB or RGB-D cameras, and odometry sensors such as wheel encoders (Yeong et al., 2021). These positioning systems have played - and continue to play - a crucial role in geomatics, mapping and environment monitoring. Integrating a positioning system with additional sensors (e.g., thermometers or pollution detectors) on a mobile platform enables the collection of spatially and temporally referenced data. When coupled with sensors capable of capturing geometric and radiometric characteristics of the surrounding environment, such as RGB cameras and LiDARs, these systems, through direct georeferencing, enable the 3D reconstruction of topographic, facility, or infrastructure-related features. Such systems, commonly referred to as mobile mapping systems, can be mounted on ground-based platforms (e.g. a wheel robot), unmanned aerial vehicles (UAVs), or aircraft. In some cases, a combination of sensors can contribute to both positioning and mapping (as in mobile laser scanning systems; Vaaja et al., 2018) while in others, specific sensors are dedicated to positioning, with the remaining ones providing geometric or colorimetric data (Toth and Grejner-Brzezinska, 2004). For instance, in LiDAR-camera systems, cameras are often used to supply colorimetric information to LiDAR data rather than participating directly in the positioning process (Vechersky et al., 2018).

1.2 Deep learning for positioning

In recent years, machine learning (ML) and deep learning (DL) have significantly advanced scientific and technological fields, including positioning systems. In particular, these techniques have played a crucial role in the development of Visual Odometry (VO) and Visual-SLAM (Simultaneous Localization and Mapping) technologies. These approaches aim to simultaneously reconstruct the environment in which an agent operates and estimate its position within it using one or more RGB cameras (Kazerouni et al., 2020). Due to their favorable balance between

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-1-W5-2025-93-2025 | © Author(s) 2025. CC BY 4.0 License.

accuracy, processing time and cost, such technologies have found widespread applications in robotics and the automotive sector.

One of the prevailing approaches in VO and Visual-SLAM involve tracking feature points across image sequences, triangulating them, and positioning new frames relative to previously triangulated points based on photogrammetric principles (Scaramuzza and Fraundorfer, 2011). This process is followed by local optimization techniques, such as sliding window bundle adjustment (BA), and global optimization methods, including global BA or pose graph optimization (Kümmerle and Grisetti, 2011). In recent years, deep learning has been progressively integrated into these frameworks to enhance specific tasks, such as tracking local features in images, up to the proposal of fully end-to-end DL-based positioning systems (Teed and Deng, 2021; Chen et al., 2023; Sarlin et al., 2023; Klenk et al., 2024; Wang et al., 2024). Several novel frameworks focus not only on real-time positioning, but also consistent large-scale mapping (Murai et al., 2024; Zhong et al., 2024).

1.3 The VAIPOSA ESA project and aim of the work

The VAIPOSA[1] project aims to investigate the integration of VO solutions with varying levels of DL components alongside traditional positioning sensors commonly used in the automotive sector. The VAIPOSA project has two main objectives:

- to assess the performance - in terms of accuracy and reliability - of DL-based components for pose estimation, particularly in challenging environments such as urban canyons or tunnels, where GNSS signals may be partially or entirely unavailable.
- to develop a non-DL-based safety mechanism capable of detecting failures in individual positioning components before the data fusion process. To ensure greater control over test environments and improve the reliability of ground truth data, the developed tools are tested within a simulated environment, enabling exhaustive verification of their performance in autonomous driving scenarios in an urban setting. The GNSS signal is simulated considering obstruction to the signal due to obstacles along the line of sight, and the position estimated in PPP (Elsheikh et al., 2023).

This paper focuses specifically on the VO component under development in the project – build upon COLMAP-SLAM (Morelli et al., 2023a), and analyses advantages and limitations of emerging DL-based approaches for vision-based positioning of moving agents. We examine deep learning ability to provide accurate and reliable solutions even in challenging scenarios, such as environments with limited geometric features, dynamic objects, poor visibility, or adverse weather conditions. The evaluation is performed using synthetically generated stereo image sequences rendered in CARLA (Dosovitskiy et al., 2017) and comparing the accuracy of estimated trajectories across different algorithms.

2. METHODOLOGY

2.1 Simulator and datasets

CARLA (Dosovitskiy et al., 2017) simulation environment leverages Unreal Engine 4[2] for rendering. To isolate the performance of the Visual Odometry algorithm, data fusion with other sensors (e.g., IMU or GNSS) is not considered. CARLA ensures a reliable ground truth generation and eliminates dependencies on external factors, such as uncertainty of stereo system calibration in terms of distortion and relative camera orientation. Multiple sequences are generated considering varying atmospheric conditions (rain, fog, etc.) and scene characterization (texture, illumination, etc.). Specifically, five case studies were examined (Figure 1). Three involved scenarios without traffic (i.e., no other vehicles present on the road in either direction): (a) daytime with clear weather, (b) nighttime with rain, and (c) nighttime with fog. The remaining two scenarios included the presence of traffic: (d) daytime with traffic, and (e) nighttime with fog and traffic.

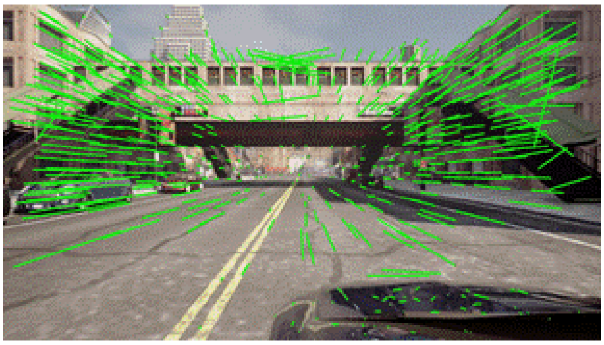


Figure 2: Example of feature tracking with ALIKE and LightGlue in COLMAP-SLAM.

2.2 Visual Odometry algorithms

Three approaches are considered for performing Visual Odometry. The selection was carried out on the basis of testing representative visual odometry methods, utilizing deep learning solutions at different levels of solving the pose estimation problem. Thus, we adopted one method using an end-to-end camera pose estimation architecture, one mixing learning-based solution for feature extraction and matching with classical camera orientation principles, and finally a baseline method, which does not incorporate any learning-based components (hand-crafted). The specific methods employed were:

- ORB-SLAM3 (Campos et al., 2021): it is an open-source SLAM solution based on hand-crafted algorithms. Image correspondences in the stream are based on ORB (Rublee et al., 2011), a computationally efficient algorithm well-suited for scenarios without significant illumination changes between consequent frames. ORB-SLAM3 is widely regarded as a benchmark in multiple research domains, including computer vision and robotics. The odometry estimation relies on a local sliding-window bundle adjustment, complemented by a global map optimization process based on pose graph optimization.
- COLMAP-SLAM (Morelli et al., 2023a): the advent of DL aimed to overcome limitations of traditional visual positioning algorithms by offering solutions that more effectively handle challenging illumination conditions. COLMAP-SLAM integrates DL methods for the extraction and tracking of image correspondences in the image stream and uses a sliding-window bundle adjustment to derive all unknowns (Morelli et al., 2023b). In our tests, image correspondences are extracted coupling ALIKED (Zhao et al., 2023), a convolutional network designed for real-time local feature extraction, with LightGlue (Lindenberger et al., 2023), a graph neural network-based matcher which identifies correspondences and filter out outliers. LightGlue is also optimized for real-time performance, employing an adaptive early-exit strategy from different convolution layers based on the network's assessment of image complexity. Figure 2 shows an example of feature tracking in a simulated urban scene. As alternative method, we also tested

[1] https://vaiposa.fbk.eu/

[2] https://www.unrealengine.com/en-US

SuperPoint (De Tone et al., 2018) combined with LightGlue. SuperPoint is a convolutional neural network trained on synthetic geometrical shapes with additional training on real images and it is considered a reference method between sparse DL-based local features.

- DROID-SLAM (Teed and Deng, 2021): it is an end-to-end DL-based SLAM framework that estimates camera poses and refined dense depth maps from a sequence of images. For each image, the network maintains two state variables: the camera pose and the inverse depth, which are updated iteratively as new frames arrive. The system builds a dynamic frame graph to represent overlapping fields of view between images and continuously updates this graph as poses and depths are refined. Thanks to that, the framework can produce meaningful results even with imperfect input depth maps, such as data coming from monocular depth estimation (MDE). The Metric3D MDE model (Yin et al., 2023) is used to derive depths from the monocular sequences. In the front-end, a convolution-based learned update operator predicts updates to both the pose and depth estimates. At the start of each iteration, dense correspondences between image pairs are computed to generate optical flow features, which are then processed to refine the state estimates. Global consistency is enforced by a Dense Bundle Adjustment layer that maps these updates into refined camera poses and pixel-level depth values, minimizing reprojection errors. For this study, as we put the focus on the odometry frameworks, we disabled the back-end optimization of DROID-SLAM, using only its front-end pose estimation with a sliding window approach.

3. RESULTS

The test trajectory spans 1.2 km and primarily consists of a closed-loop path with an approximately squared shape (150 meters per side). The five tested scenarios (Figure 1) depict an urban environment with rich textures, no urban canyoning, various and traffic conditions and illumination or weather changes. The simulated stereo system comprises two cameras with identical orientations, mounted 1.40 meters apart on the vehicle's roof and aligned with the vehicle's direction of motion. The camera in CARLA is modelled using the pinhole camera model, with the principal point located at the centre of the image and no radial or other distortions. This ideal imaging setup, free from noise and distortion modelling errors, provides an opportunity to evaluate the accuracy of the odometry modules under analysis without any effects from miscalibration. Furthermore, the use of CARLA simulator enables generating a reliable ground truth of the stereo camera poses. The autonomous agent followed a pre-planned route, maintaining an approximate speed of 50 km/h. The five sequences are composed of 858 stereo pairs acquired at 5 fps.

For the quantitative evaluation, two metrics are considered: the Absolute Pose Error (APE) and the Relative Pose Error (RPE), both computed using the Evo library (Grupp, 2017). APE and RPE are calculated by aligning the trajectory estimated with each SLAM algorithm with the GT trajectory via a Helmert transformation based on all estimated camera positions.

Table 1 shows the RMSE of APE and RPE comparing the estimated poses with the ground truth (GT) trajectories provided by the simulator. Figure 4 reports APE box plot subdivided by dataset, to easily compare each method performance by sequence. Figure 5 reports instead APE box plot subdivided by V-SLAM algorithm, showing how each SLAM performs across the different sequences. Figures 6 shows, for each dataset, trajectories and RPEs in time. Due to the inflation of the metrics by scale issues of DROID-SLAM, in the plots we included only results of DROID-SLAM with the corrected scale factor. In the evaluation, loop closure detection was not utilized and explicitly disabled where available (ORB-SLAM and DROID-SLAM), as its occurrence is unlikely in typical automotive scenarios and would artificially compensate for accumulated drift. Nevertheless, a closed, approximately square trajectory was selected for evaluation, as it facilitates clearer visualization of both translational and rotational drift.

3.1 ORB-SLAM3 without loop closure detection

The firsts tests with ORB-SLAM3 showed frequent loss of ORB feature tracking with default configuration, which necessitates repeated reinitialization of the SLAM process (Figure 3). This tracking failure may be attributed to the vehicle's speed and the 5 fps frame rate of the image stream, which likely causes significant changes in the appearance of local features, making them challenging for ORB to track effectively. Fine-tuning specific configuration parameters related to the extraction of ORB features enabled the complete trajectory to be reconstructed without discontinuities. The parameters that have shown the highest impact on the results were: the number of ORB features (*f*) extracted and the number of levels (*levels*) in the scale pyramid.

(a) No traffic, day	RMSE APE	RMSE RPE
ORB-SLAM3	0.339	0.014
COLMAP-SLAM-ALIKED	0.874	**0.005**
COLMAP-SLAM-SuperPoint	**0.570**	0.006
Droid-slam unscaled	10.947	0.164
Droid-slam scaled	7.283	0.074

(b) No traffic, rainy night	RMSE APE	RMSE RPE
ORB-SLAM3	1.801	0.045
COLMAP-SLAM-ALIKED	0.815	**0.007**
COLMAP-SLAM-SuperPoint	**0.714**	**0.007**
Droid-slam unscaled	16.497	0.230
Droid-slam scaled	10.460	0.094

(c) No traffic, foggy night	RMSE APE	RMSE RPE
ORB-SLAM3	**0.787**	0.016
COLMAP-SLAM-ALIKED	0.819	**0.008**
COLMAP-SLAM-SuperPoint	1.610	0.011
Droid-slam unscaled	14.951	0.195
Droid-slam scaled	10.766	0.095

(d) Traffic, day	RMSE APE	RMSE RPE
ORB-SLAM3	**0.300**	0.015
COLMAP-SLAM-ALIKED	0.486	**0.006**
COLMAP-SLAM-SuperPoint	0.647	0.011
Droid-slam unscaled	10.178	0.144
Droid-slam scaled	7.547	0.063

(e) Traffic, foggy night	RMSE APE	RMSE RPE
ORB-SLAM3	-	-
COLMAP-SLAM-ALIKED	19.471	1.826
COLMAP-SLAM-SuperPoint	**11.445**	**0.110**
Droid-slam unscaled	35.671	0.346
Droid-slam scaled	35.668	0.349

Table 1: Absolute Pose Error (APE) and Relative Pose Error (RPE) in meters for the five test datasets. The lowest values in green.

The number of ORB features extracted is inextricably linked with the number of levels in the scale pyramid: specifically, increasing the latter allows to perform a finer tuning on the former. Moreover, since pyramid level values larger than 10 have exhibited no sensible improvement to the results, and values lower than 8 have shown a consistent loss of tracking, especially in sharp curves, only values of 10 and 8 have been used. Finally, number of ORB features larger than 2000 have shown an increased divergence from GT or loss of tracking, especially in curves, while values below 700 have shown loss of tracking in curves and in sequences with significant lack of visibility and adverse weather conditions.

Variations of the pyramid scale factor, and the initial and minimum threshold for detected corners have no significant impact on the results, therefore they have been left at the default values for all analyses (*scaleFactor*=1.2, *minThFAST*=7, *iniThFAST*=20). No single parameter combination was found to generalize well across all datasets, and any small change in *f* and number of *levels* leads to a tracking loss. Table 1 reports the results for ORB-SLAM with the best parameters that allowed to have a complete trajectory. This highlights the strong sensitivity of ORB features to parameter settings and suggests that either real-time parameter tuning strategies must be implemented, or the potential for tracking failure - and thus the need to reinitialize the SLAM system - must be accounted for.

Despite this sensitivity, ORB-SLAM3 consistently achieved an average APE across the four datasets (a-d) between 0.3 and 1.8 m, i.e. less than 1.5‰ error. This small value can be attributed to the ideal pinhole camera model without any errors in the estimation of camera distortions. In the foggy-night traffic scenario, the worst scenario in terms of visibility, ORB-SLAM3 failed to recover the full trajectory, even with fine tuning of the afore-mentioned critical parameters. In terms of relative pose error, RMSE RPE is at centimeter level, ranging from 1.4 to 4.5 cm.

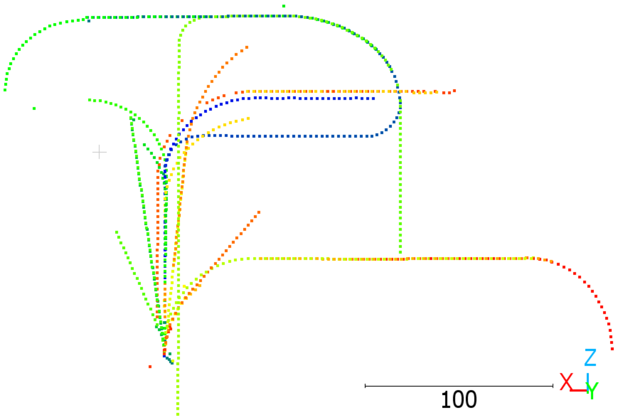


Figure 3: Trajectory estimation with ORB-SLAM 3 with default parameters. Colors show the timestamps (from blue through green to red).

3.2 COLMAP-SLAM

In the COLMAP-SLAM solution, ALIKED and SuperPoint combined with LightGlue are utilized for both feature tracking and mapping. An example of feature tracking in COLMAP-SLAM is reported in Figure 2. The average optical flow is used for keyframe selection, therefore only frames showing enough change in the scene appearance are considered as keyframes (5 px threshold). Differently from ORB-SLAM, the initial tests revealed continuity in feature tracking, enabling the estimation of a single trajectory across the five test datasets without specific parameter fine tuning. As shown in Table 1 and Figure 4, COLMAP-SLAM achieved APE in the range of 0.57 and 1.61 m, i.e., less than 1.3‰ error. It managed to orient all the keyframes also in the most challenging scenarios (e), *traffic, foggy night*, even if a quite high RMSE APE of 19.47 m for ALIKE and 11.44 m for SuperPoint. Even if APE errors are quite high, the relative error RPE for SuperPoint (11 cm) has the same order of magnitude of the other datasets, while the RPE of ALIKED is significantly higher because of an outlier.

Although COLMAP-SLAM demonstrates continuous tracking and produces a seamless trajectory across all test cases, in the initial tests with both ALIKED and SuperPoint feature extractors, it exhibited significant performance degradation when the number of keypoints was limited to a maximum of 1000. When no such constraint is applied - resulting in approximately 3000 features per image - the APE improves substantially up to two times, therefore in the results are reported only for the scenarios where max number of features are extracted.

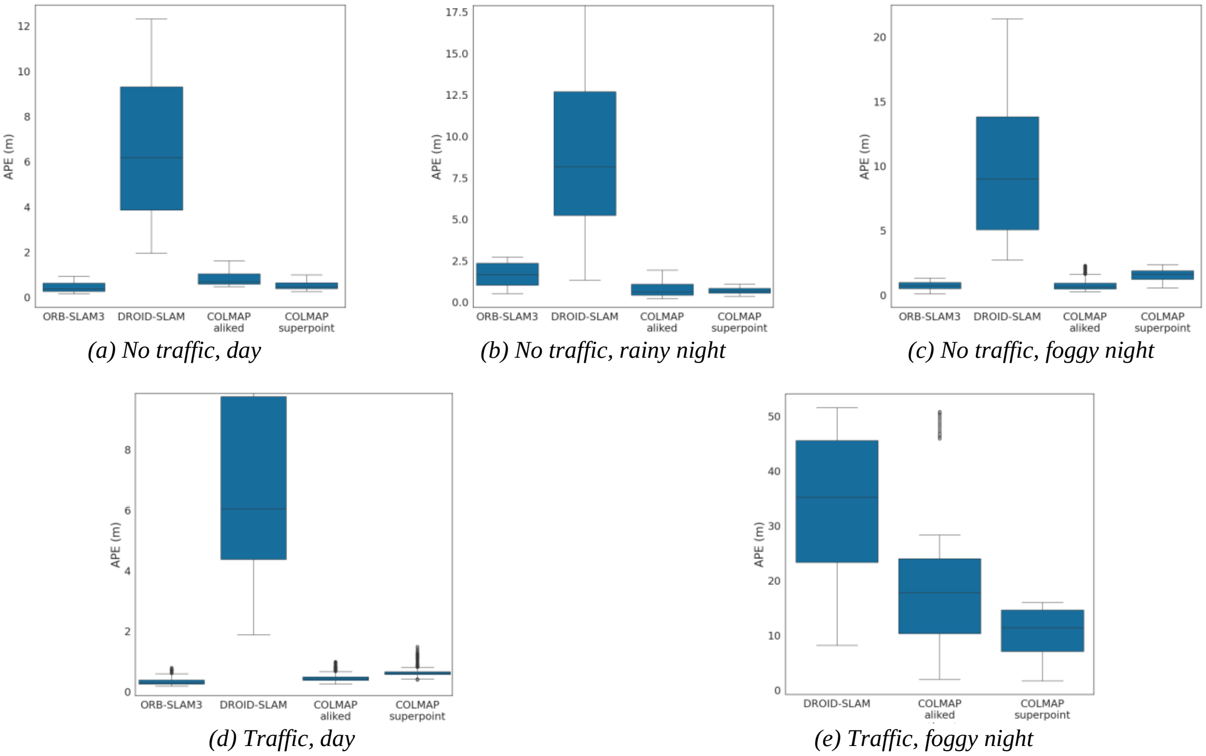
3.3 DROID-SLAM front-end

Compared to ORB-SLAM3, DROID-SLAM successfully maintains a continuous trajectory throughout all tested image sequences. However, it exhibits more pronounced drift accumulation than COLMAP-SLAM, with an APE RMSE significantly bigger both with and without scaling factor re-estimation This substantially lower accuracy compared to COLMAP-SLAM is partially attributable to DROID-SLAM’s reliance on a monocular camera and a neural network-based approach for monocular depth estimation (Yin et al., 2023). The decrease in APE error is also reflected by the RPE that it consistently at least ten times worse with respect to ORB-SLAM3 and COLMAP-SLAM. Thus, since DROID-SLAM uses a sliding window optimization approach, limiting the number of wrongly oriented frames through outlier detection could potentially improve its reliability and accuracy. Nevertheless, DROID-SLAM produced smooth and topologically correct trajectories in all test cases, including the most challenging dynamic night scenario.

As the DROID-SLAM pose estimation is based on a mixture of RGB and prior depth information, a monocular depth prediction was performed. Due to that, the resulting trajectory is not properly scaled. Although the issue could be theoretically solved by employing a dense stereo depth estimation, initial trials with using such a model did not result in a topologically correct trajectory. A potential cause of such phenomenon could be that the original weights of DROID-SLAM, pretrained strictly on simulated sequences with monocular cameras, do not generalize well to the dense stereo metric input.

3.4 Discussion

Five synthetic datasets were generated using CARLA to allow full control over illumination, weather conditions, and the presence or absence of dynamic objects (e.g., traffic), while also providing access to reliable ground truth data. A comparison of APE performance indicates that both ORB-SLAM and COLMAP-SLAM are viable alternatives. However, ORB-SLAM requires careful parameter tuning for ORB feature extraction to maintain robust tracking; otherwise, frequent reinitialization of the SLAM system may be necessary. DROID-SLAM and COLMAP-SLAM are capable of handling even the most challenging sequence (e) *traffic-foggy-dark* without losing tracking, though with generally higher error compared to other sequences. Notably, in most cases, DROID-SLAM exhibits an RMSE APE at least an order of magnitude higher than that of ORB-SLAM and COLMAP-SLAM, showing that there is space of improvements for end-to-end DL SLAM approaches.



(a) No traffic, day

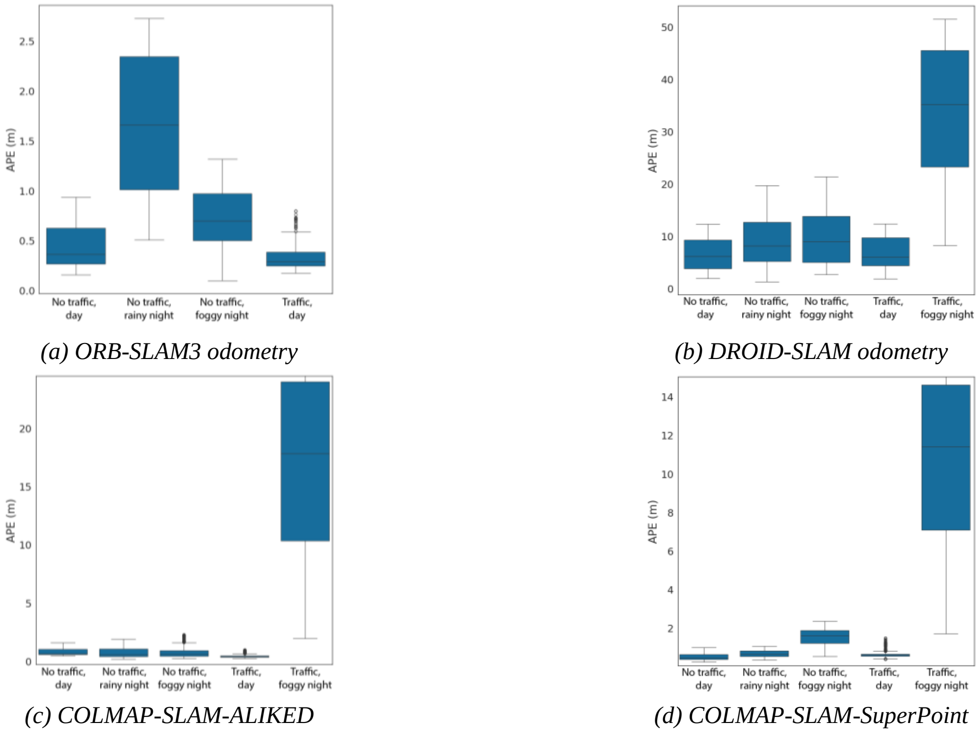
(b) No traffic, rainy night

(c) No traffic, foggy night

(d) Traffic, day

(e) Traffic, foggy night

Figure 4: Distributions of APE for all examined VO methods group by the test scenario (a – e).



(a) ORB-SLAM3 odometry

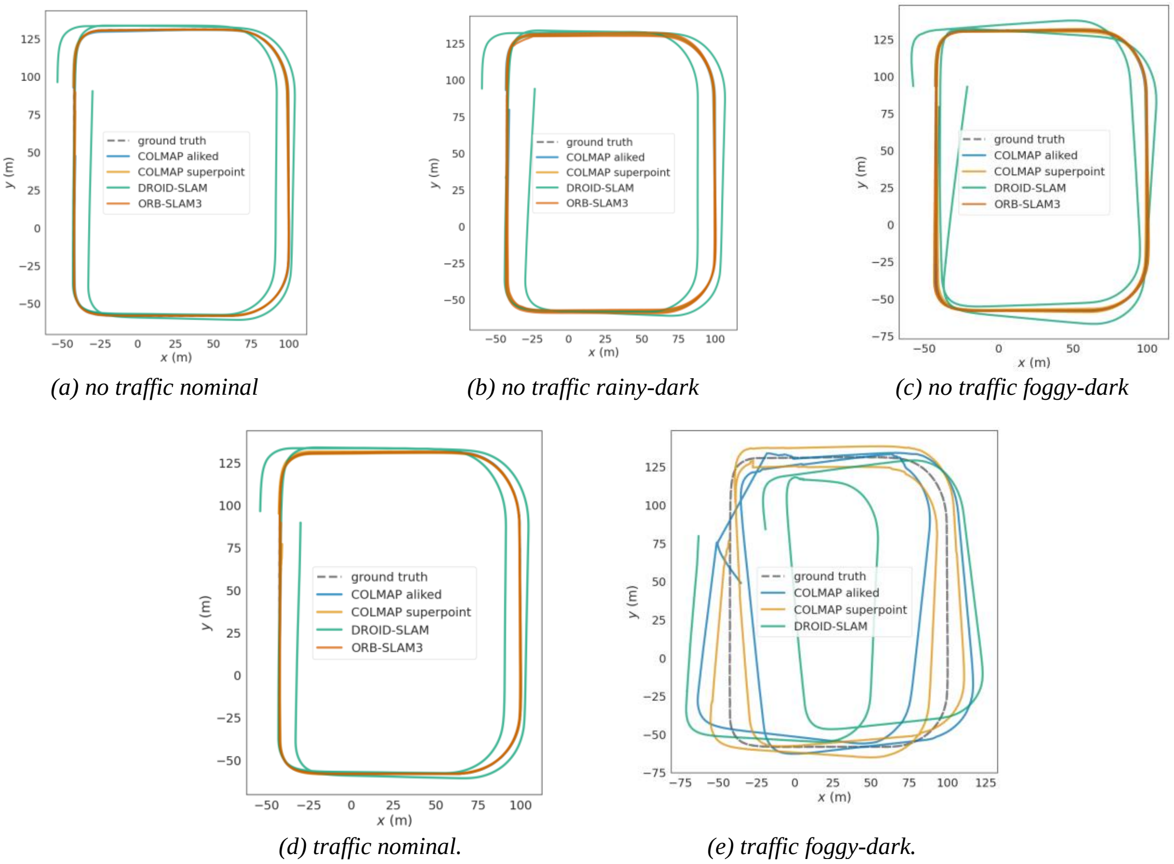
(b) DROID-SLAM odometry

(c) COLMAP-SLAM-ALIKED

(d) COLMAP-SLAM-SuperPoint

Figure 5: Distributions of APE for all successfully processed test sequences grouped by the VO module (a – d).

On the other hand, the performance of DROID-SLAM is noteworthy, given that it is based on monocular depth estimation. Figure 5 illustrates the performance of each method across the different datasets. The results indicate that the presence of traffic - although not particularly dense - does not substantially degrade APE performance under favourable lighting conditions. This is likely due to the abundance of visual features, which enables the exclusion of tie points associated with moving objects during the RANSAC process. In contrast, when traffic is combined with nighttime lighting and fog, the scarcity of features impairs the



(a) no traffic nominal *(b) no traffic rainy-dark* *(c) no traffic foggy-dark*

(d) traffic nominal. *(e) traffic foggy-dark.*

Figure 6: Planar projection of the trajectories and RPE in time for every tested VO solution for all sequences (a – e).

ability to filter out tie points of dynamic objects, resulting in a significant increase in APE. In the absence of traffic, sequences with nighttime lighting rain and fog (b and c) consistently yield worse performance compared to the reference datasets, i.e., daylight without traffic scenario (a) and daylight with traffic (d).

4. CONCLUSIONS

This study presents a comparative evaluation of three SLAM algorithms - ORB-SLAM3, COLMAP-SLAM, and DROID-SLAM - within a realistic urban simulation environment designed to reflect typical automotive operating conditions. The results indicate that while ORB-SLAM3 and COLMAP-SLAM achieve the best absolute pose error (APE) across most scenarios, ORB-SLAM3 is highly sensitive to configuration parameters and prone to tracking failures unless finely tuned. COLMAP-SLAM offers a robust alternative, demonstrating consistent feature tracking keeping accurate trajectory estimation also in challenging conditions, albeit with higher APE in the most difficult scene (traffic with nighttime light condition and fog). DROID-SLAM, though capable of maintaining continuous tracking, suffers from significant drift and reduced accuracy, likely due to its monocular input and learned depth estimation strategy. Overall, the reported findings underscore the importance of matching SLAM system design choices - feature type, configuration flexibility, and sensor modality.

In future work, testing will be expanded to include longer trajectories and more diverse environments, with a particular focus on urban canyon scenarios. Additionally, these visual odometry algorithms will be integrated with other simulated sensors, such as wheel odometry, inertial measurement units, and global navigation satellite systems.

ACKNOWLEDGEMENTS

This work has been partially supported by the European Space Agency (ESA) VAIPOSA project, funded under the NAVISP-EL1-087 programme, agreement 4000145024/24/NL/AK/kg.

REFERENCES

Aziz, T., Koo, I., 2025. A Comprehensive Review of Indoor Localization Techniques and Applications in Various Sectors. *Applied Science*, 15, 1544.

Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M. and Tardós, J.D., 2021. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, *37*(6), pp. 1874-1890.

Chen, C., Wang, B., Lu, C. X., Trigoni, N., Markham, A., 2023. Deep learning for visual localization and mapping: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(12), pp. 17000-17020.

Correa, A., Barcelo, M., Morell, A. and Vicario, J.L., 2017. A review of pedestrian indoor positioning systems for mass market applications. *Sensors*, *17*(8), pp. 1927.

DeTone, D., Malisiewicz, T. and Rabinovich, A., 2018. Superpoint: Self-supervised interest point detection and description. Proc. *CVPR*, pp. 224-236.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A. and Koltun, V., 2017, October. CARLA: An open urban driving simulator. Proc. *Conference on robot learning,* pp. 1-16.

Elsheikh, M., Iqbal, U., Noureldin, A. and Korenberg, M., 2023. The implementation of precise point positioning (PPP): a comprehensive review. *Sensors*, *23*(21), pp. 8874.

Fayyad, J., Jaradat, M.A., Gruyer, D. and Najjaran, H., 2020. Deep learning sensor fusion for autonomous vehicle perception and localization: A review. *Sensors*, *20*(15), pp. 4220.

Grupp, M., 2017. *evo: Python package for the evaluation of odometry and slam* [online]

Kazerouni, I.A., Fitzgerald, L., Dooly, G. and Toal, D., 2022. A survey of state-of-the-art on visual SLAM. *Expert Systems with Applications*, *205*, pp. 117734.

Klenk, S., Motzet, M., Koestler, L., Cremers, D., 2024. Deep event visual odometry. Proc. *International Conference on 3D vision (3DV)*, pp. 739-749.

Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K. and Burgard, W., 2011, May. g2o: A general framework for graph optimization. Proc. *ICRA,* pp. 3607-3613.

Lindenberger, P., Sarlin, P.E. and Pollefeys, M., 2023. Lightglue: Local feature matching at light speed. In: *Proc. ICCV,* pp. 17627-17638.

Morelli, L., Ioli, F., Beber, R., Menna, F., Remondino, F. and Vitti, A., 2023a. COLMAP-SLAM: A framework for visual odometry. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *48,* pp. 317-324.

Morelli, L., Menna, F., Vitti, A., Remondino, F. and Toth, C., 2023b. Performance Evaluation of Image-Aided Navigation with Deep-Learning Features. *Proceedings ION GNSS+*, pp. 2048-2056.

Murai, R., Dexheimer, E., Davison, A. J., 2024. MASt3R-SLAM: Real-Time Dense SLAM with 3D Reconstruction Priors. *arXiv preprint arXiv:2412.12392*.

Rublee, E., Rabaud, V., Konolige, K. and Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. *Proc. ICCV,* pp. 2564-2571.

Sanguino, T.D.J.M., 2017. 50 years of rovers for planetary exploration: A retrospective review for future directions. *Robotics and Autonomous Systems*, *94*, pp. 172-185.

Sarlin, P. E., DeTone, D., Yang, T. Y., Avetisyan, A., Straub, J., Malisiewicz, T., Bulo, S. R., Newcombe, R., Kontschieder, P., Balntas, V., 2023. Orienternet: Visual localization in 2d public maps with neural matching. Proc. *CVPR,* pp. 21632-21642.

Scaramuzza, D. and Fraundorfer, F., 2011. Visual odometry [tutorial]. *IEEE robotics & automation magazine*, *18*(4), pp. 80-92.

Teed, Z. and Deng, J., 2021. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, *34*, pp. 16558-16569.

Toth, C. and Grejner-Brzezinska, D., 2004. Redefining the paradigm of modern mobile mapping. *Photogrammetric Engineering & Remote Sensing*, *70*(6), pp. 685-694.

Vaaja, M., Kurkela, M., Maksimainen, M., Virtanen, J.P., Kukko, A., Lehtola, V.V., Hyyppä, J. and Hyyppä, H., 2018. Mobile mapping of night-time road environment lighting conditions. *Photogrammetric Journal of Finland*, *26*(1).

Vechersky, P., Cox, M., Borges, P., and Lowe, T., 2018. Colourising point clouds using independent cameras. *IEEE Robotics and Automation Letters*, *3*(4), pp. 3575-3582.

Wang, K., Zhao, G., Lu, J., 2024. A deep analysis of visual SLAM methods for highly automated and autonomous vehicles in complex urban environment. *IEEE Transactions on Intelligent Transportation Systems*, 25(9), pp. 10524-10541.

Weyler, J., Läbe, T., Magistri, F., Behley, J. and Stachniss, C., 2023. Towards domain generalization in crop and weed segmentation for precision farming robots. *IEEE robotics and automation letters*, *8*(6), pp. 3310-3317.

Yeong, D.J., Velasco-Hernandez, G., Barry, J. and Walsh, J., 2021. Sensor and sensor fusion technology in autonomous vehicles: A review. *Sensors*, *21*(6), pp. 2140.

Yin, W., Zhang, C., Chen, H., Cai, Z., Yu, G., Wang, K., Chen X. and Shen, C., 2023. Metric3d: Towards zero-shot metric 3d prediction from a single image. Proc. *ICCV,* pp. 9043-9053.

Yurtsever, E., Lambert, J., Carballo, A. and Takeda, K., 2020. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, *8*, pp. 58443-58469.

Wen, B., Trepte, M., Aribido, J., Kautz, J., Gallo, O., Birchfield, S., 2025. FoundationStereo: Zero-Shot Stereo Matching. *Proc. CVPR*.

Zhao, X., Wu, X., Chen, W., Chen, P.C., Xu, Q. and Li, Z., 2023. Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. *IEEE Transactions on Instrumentation and Measurement*, *72*, pp. 1-16.

Zhong, X., Pan, Y., Behley, J., Stachniss, C., 2023. Shine-mapping: Large-scale 3d mapping using sparse hierarchical implicit neural representations. Proc. *ICRA*, pp. 8371-8377.