

Realism-Oriented Design, Verification, and Validation of Novel Robust Navigation Solutions [†]

Sorin Andrei Negru ¹, Patrick Geragersian ¹, Ivan Petrunin ^{1,*} , Raphael Grech ² and Guy Buesnel ²

¹ School of Aerospace, Transport, and Manufacturing, Cranfield University, Bedfordshire MK43 0AL, UK; sorin.negru@cranfield.ac.uk (S.A.N.); patrick.geragersian@cranfield.ac.uk (P.G.)

² Spirent Communications PLC, West Sussex RH10 1BD, UK; raphael.grech@spirent.com (R.G.); guy.buesnel@spirent.com (G.B.)

* Correspondence: i.petrunin@cranfield.ac.uk

[†] Presented at the European Navigation Conference 2023, Noordwijk, The Netherlands, 31 May–2 June 2023.

Abstract: Urban environments are characterized by a set of conditions underpinning degradation Position, Navigation and Timing (PNT) signals, such as multipath and non-line of sight (NLOS) effects, negatively affecting the position and the navigation integrity during the Uncrewed Aerial Vehicles (UAVs) operations. Before the deployment of such uncrewed aerial platforms, a realistic simulation set-up is required, which should facilitate the identification and mitigation of the performance degradation that may appear during the actual mission. This paper presents the case study of the development of a robust Artificial Intelligence (AI)-based multi-sensor fusion framework using a federated architecture. The dataset for this development, comprising the outputs of a Global Navigation Satellite System (GNSS) receiver, an Inertial Measurement Unit (IMU) and a monocular camera is generated in a high-fidelity simulation framework. The simulation framework is built around Spirent's GSS7000 simulator, software tools from Spirent (SimGEN and SimSENSOR) and OKTAL-SE (Sim3D), where the realism for the vision sensor data generation is provided by a photorealistic environment generated using the AirSim software with the Unreal Engine aid. To verify and validate the fusion framework a hardware in the loop (HIL) set-up has been implemented using the Pixhawk controller. The results obtained demonstrate that the presented HIL set-up is the essential component of a more robust navigation solution development framework, providing resilience under conditions of GNSS outages.



Citation: Negru, S.A.; Geragersian, P.; Petrunin, I.; Grech, R.; Buesnel, G.

Realism-Oriented Design, Verification, and Validation of Novel Robust Navigation Solutions. *Eng. Proc.* **2023**, *54*, 57. <https://doi.org/10.3390/ENC2023-15424>

Academic Editors: Tom Willems and Okko Bleeker

Published: 29 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: realistic simulation; sensor fusion; machine learning; navigation integrity

1. Introduction

The world of uncrewed and autonomous systems is quickly evolving in all sectors, bringing a new set of challenges for platform positioning and navigation solutions from both performance and certification points of view. Considering tighter requirements for positioning accuracy and integrity from the emerging use cases, such as drone operations in urban environments and BVLOS (beyond-visual line-of-sight conditions), extensive use of AI in positioning and fusion engines, repeatable and verifiable testing, and validation of the navigation systems gains significant importance.

As can be seen from [1,2], AI models can be used to increase the navigation robustness of UAVs under multipath and NLOS conditions [3]. Moreover, given the complexity that such navigation solutions must be performing in, a realistic yet repeatable development, testing, and validation environment is required. Different realistic environments were developed previously as can be seen in [4,5] with the aid of Unreal Engine, a photorealistic simulation software, and AirSim (Aerial Informatics and Robotics Simulation) [6], being then deployed directly on an embedded computer, to test, in real-time, the required computational load. This enables the possibility of introducing into the photorealistic environment

optical sensors, such as monocular or stereo cameras, or even Lidars, leading to additional sources of positioning information. AirSim, a plug-in compatible with Unreal Engine, also allows the possibility to model the intrinsic and distortion parameters of the optical camera, mounted on the desired aerial or ground platforms. Moreover, lighting effects based on predefined daytime hours can be added, increasing the realism of the environment. In this way, it is possible to deploy visual algorithms more easily, as described in [7], where AirSim is used for the fast prototyping of algorithms used by racing drones, in a photorealistic environment.

Although it is possible to validate algorithms directly on embedded Companion Computers (CCs) before actual deployment in the field, external factors such as multipath, NLOS, and IMU stochastic and deterministic errors are not considered in the simulation environment. Multipath in urban canyons plays a disruptive force when acquiring GNSS location. Due to the obscurations from buildings, NLOS signals reduce positioning accuracy and increase uncertainty in the receiver location. Furthermore, IMU biases and noises lead to the divergence of estimated position and ground truth. This effect is known as random walk (RW). Therefore, the simulation of such physical effects is crucial when evaluating potential algorithms that can be deployed in such environments. Considering the challenges in modeling IMU and GNSS sensors in virtual urban environments, more realistic sensor-based modeling is required.

The considered case study is testing the performances of an AI-based multi-sensor positioning solution, using a federated fusion architecture, with GNSS, INS (Inertial Navigation System), and a monocular camera, in a photorealist environment. The paper is organized as follows: in Section 2, the federated fusion framework is presented along with all the sensors used; in Section 3, the hardware in the loop (HIL) configuration with all the specifications of the set-up is presented; in Section 4, it is demonstrated that the proposed testing and validation system enables realism in simulation and repeatability analysis concerning variability in noise and satellite occlusion and comprehensively covers needs in the verification of AI-based navigation algorithms; and, finally, in Section 5, the conclusions are presented.

2. Fusion Framework

Global Navigation Satellite Systems (GNSS) is a constellation of satellites that transmit signals to ground-based receivers. These signals are used to determine the receiver's position, velocity, and time, which are critical components for navigation and timing applications. The most widely used GNSS system is the Global Positioning System (GPS), which was developed by the United States Department of Defense. Other GNSS systems include the Russian GLONASS, the European Galileo, and the Chinese BeiDou.

One of the major disadvantages of GNSS is its performance in urban canyons. Urban canyons refer to areas with tall buildings and narrow streets, which can create obstructions that block or reflect GNSS signals. This can result in degraded signal quality and increased errors in position determination. In urban canyons, the signals can be reflected off buildings or blocked entirely, leading to multipath errors, where the receiver receives multiple signals of the same satellite at different times and angles, causing confusion and inaccuracies in positioning. Additionally, the signal can be absorbed or weakened by atmospheric conditions or other sources of interference, such as radio waves or electromagnetic fields.

To mitigate the effects of urban canyon environments, GNSS receivers can use a variety of techniques such as multi-constellation and multi-frequency signal reception, as well as inertial measurement sensors to estimate position and velocity when GNSS signals are blocked or unavailable. However, these techniques may not always be sufficient in highly obstructed urban environments, highlighting the need for alternative positioning technologies in some scenarios. Considering that IMU sensors based on MEMS (Micro-Electro-Mechanical System) technology are the source of large time-dependent navigation errors under long GNSS outages, visual sensors can be considered for integration as a more resilient solution. VO (Visual Odometry) algorithms can be used to estimate the camera

ego-motion, providing alternative relative positioning information. Considering that urban and sub-urban environments are characterized by many details, for this framework, an ORB (Oriented FAST and rotated BRIEF) feature detector algorithm has been implemented because of its performance discussed in [8].

To fuse all the data gathered from the sensors described above, a centralized sensor fusion framework can be implemented potentially. However, the required computational load decreases the overall efficiency, leading to the so-called ‘dimensional disaster effect’ [9]. Instead, a decentralized fusion framework requires less computational power, allowing an easier integration on robotic platforms, for real-time applications. In these terms, the paper implements a decentralized federated sensor fusion framework as described in [2], using local filters in the form of two EKFs (Extended Kalman Filters) and a GRU (Gated Recurrent Unit) as the master filter.

GRUs are a variation of RNNs that address the vanishing gradient problem by using gating mechanisms to selectively pass information forward and backward through time. The gating mechanisms in GRUs include an update gate and a reset gate, which control how much of the current input is added to the current state and how much of the previous state is forgotten, respectively. The mechanics of which are shown in Figure 1b.

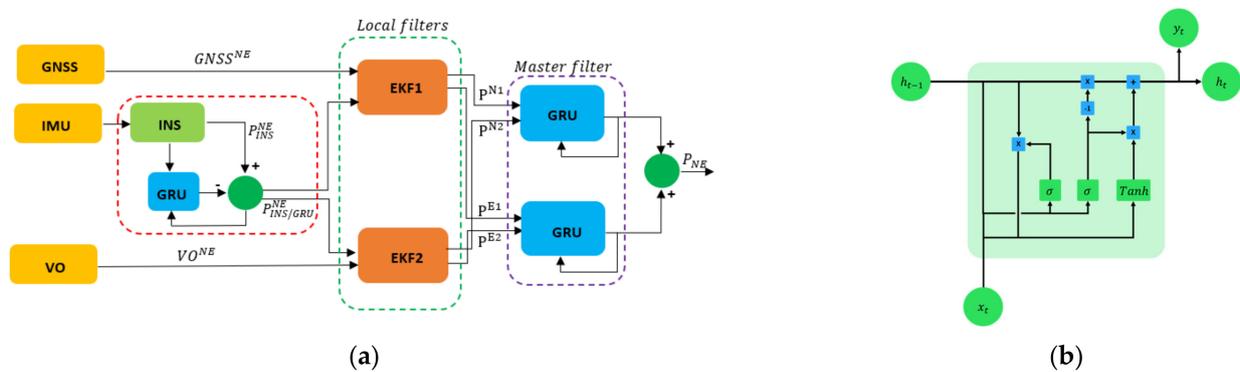


Figure 1. Federated fusion architecture (a) [2]; GRU unit (b).

GRUs have become popular for a variety of applications, including natural language processing, speech recognition, and time series prediction. They have been shown to be effective at capturing long-term dependencies in sequential data and are often used as baseline architecture in comparison to more complex models. Overall, GRUs offer a flexible and effective tool for modeling sequential data and are an important component of modern deep-learning architectures.

For reducing Inertial Navigation System (INS) position and velocity errors, the paper additionally proposes the utilization of Inertial Measurement Unit (IMU) reading prediction by using a deep neural network, utilizing GRU, to compensate for the aforementioned errors, as it can be seen in the first left part of Figure 1. Specifically, the GRU-based error correction model is trained on a dataset composed of IMU readings, where raw accelerometer and gyroscope measurements serve as input features, and the distance error between the INS-estimated position and ground truth data constitutes the corresponding output.

To obtain position data from the accelerometer and gyroscope sensors, Direction Cosine Matrix (DCM) transformation is employed to convert these sensor readings from a body frame to the navigational NED frame. Velocity and position data are subsequently derived through integration, as presented in [10]. However, it should be noted that the integration process of noise and bias from the sensors generates a random walk effect that diminishes the accuracy of INS information over extended time intervals.

To mitigate this issue, a GRU network is proposed to estimate the error engendered by the mathematical process of position information generation. By doing so, the proposed

approach enhances the accuracy of INS-derived information over extended periods of time, without requiring frequent updates from Global Navigation Satellite Systems (GNSS).

Thereby, the first EKF is fusing GNSS and IMU positioning data, with the aid of a GRU block, in a loosely coupled scheme, while the second EKF is fusing the same corrected IMU data, but with the VO algorithm aid, leading to a VIO (Visual Inertial Odometry) framework. Further, the outputs from the two EKFs are processed by a trained master GRU block as presented in [2]. To decrease the computational load required by the master GRU, the main block, is divided into two sub-blocks, each processing only one coordinate, the N (North) and, respectively the E (East) coordinate.

3. Hardware in the Loop Configuration

Most of the drones are equipped with a CC (Companion Computer) usually used to process data, from external sensors, such as monocular cameras, or to deploy AI-based models to support the Flight Control Unit (FCU) computational cost, which is responsible mainly for the navigation, control, and stability part of the UAV. An HIL simulation is performed to validate the sensor fusion framework presented in the previous section. Thus, the first step to bring realism into the simulation set-up is to implement a Pixhawk 2.4 board as an FCU that is directly connected to a local computer and used as a CC. With the aid of Unreal Engine and AirSim, it is possible to link the FCU to a photorealistic environment. Using photogrammetry data in Unreal Engine, it is possible to mimic the nature of urban environments, increasing the level of detail in terms of light intensity, realistic weather effects, and material properties. The main role of the photorealistic environment is to support validation, and tests of optical sensors that the UAV is equipped with, being able to specify camera intrinsic and distortion parameters. For the presented scenario, photogrammetry data from Toulouse and San Francisco are used, as can be seen in Figure 2.

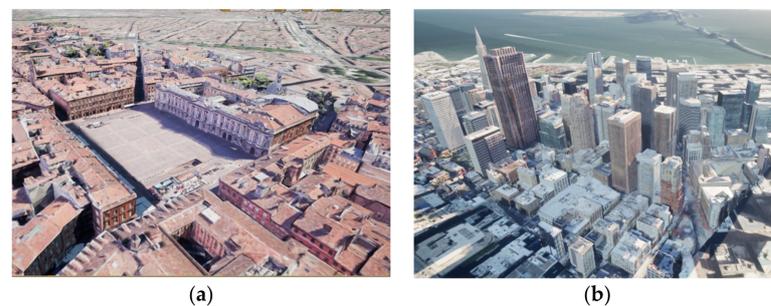


Figure 2. The 3D environment model for Toulouse (a); 3D environment model for San Francisco (b).

Once the link between the FCU, AirSim, and, respectively, Unreal Engine, is established, the trajectory commands generated from a Python file that is linked to the AirSim plug-in are transmitted further to the Spirent GSS7000 simulator, as can be seen in Figure 3. This allows performing HIL tests in an iterative way, under the same trajectory assumption. At the same time, the estimated trajectory from the monocular camera is stored in an Excel file.

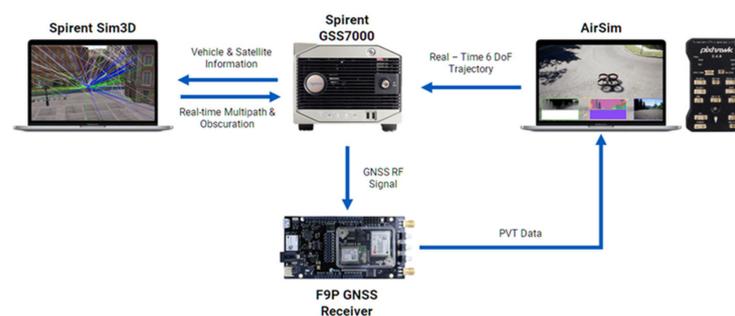


Figure 3. HIL set-up.

Thus, to assess the effectiveness of the proposed architecture, it is necessary to conduct training, utilizing realistic scenarios obtained through the use of the Spirent GSS7000 simulator and SimSENSOR obtained from Spirent PLC in Paignton, United Kingdom. Spirent GSS7000 and SimGEN simulate GNSS Radio Frequencies (RF) that are generated by satellites in each constellation. Simulation time and constellation settings can be changed to improve realism by conducting time-accurate comparisons. Furthermore, due to the repeatability of the simulation set-up, this allows for direct comparisons of different algorithm techniques to be evaluated, reducing independent variables for testing. These signals are then fed into a GNSS receiver for capturing positioning data. This enables the testing of Hardware-in-the-Loop (HIL) simulation which improves scenario realism. To produce IMU data required for training, SimSENSOR is used to generate accelerometer and gyroscope measurements that the vehicle is exhibiting. SimSENSOR allows for the simulation of realistic sensor errors by changing the SimSENSOR deterministic and stochastic error parameters such as random walk rate and bias offset. This adds an additional layer of realism to the simulation scenario. For the simulation in this paper, IMU stochastic and deterministic errors were tuned to mimic the characteristics of an Advanced Navigation Orientus IMU sensor, the specifications of which are listed in Table 1. These data are then transmitted via the UDP communication protocol to the AirSim simulator.

Table 1. INS sensor and GNSS receiver specifications.

Sensor Specifications					
Accelerometer		Gyroscope		U-Blox F9P GNSS receiver	
Scaling factor (ppm)	500	Scaling factor (ppm)	500	Pseudo-range accuracy (m)	3
Bias (mg)	0.1	Bias (deg/h)	0.001	Pseudo-range rate accuracy (m/s)	0.5
ARW (m/s/sqrt(h))	0.003	GRW (deg/sqrt(h))	0.003	Update rate (Hz)	1
Update rate (Hz)	100			Constellations	GPS L2C, GLO L2OF, GAL E5b, BDS B2I, QZSS L2C

Data collected from the sensors and simulation systems were stored in a CSV file format, with the time and date of each timestep recorded. The data were subsequently processed in MATLAB to generate the ground truth data that served as the training target. The input data consisted of the raw output from the IMU sensors (accelerometer and gyroscope in XYZ) and the GNSS latitude, longitude, and height information recorded at a rate of 1 Hz. Using interpolation, these data were synchronized to the same timestep as the accelerometer and gyroscope data, which were recorded at 100 Hz.

To further improve realism regarding urban canyon scenarios, OKTAL-SE Sim3D software is used to introduce the multipath effect on GNSS signals using the ray-tracing method. The program takes into consideration the specific geometry in the desired urban environments that are simulated in this paper. Furthermore, the different material properties underpinning the signal reflection characteristics are also considered for an additional layer of realism. This will generate line-of-sight, multipath, and non-line-of-sight conditions for the GNSS signals that are processed by the receiver. No additional multipath mitigation techniques other than those already provided by the receiver manufacturer are used to aid in these conditions.

4. Evaluation

To evaluate the proposed sensor fusion framework, Toulouse, and San Francisco, were chosen to validate the HIL configuration by adding diversity in the evaluation process.

The fusion framework performances for San Francisco can be observed in the table below. Two cases were considered, one with the corrected INS data with the aid of a GRU block, and another one, without. It can be observed that the output of the first extended Kalman filter, fusing data from the GNSS and the IMU without the GRU aid, does have inferior positioning performances compared to the one with the GRU aid leading to a horizontal position RMSE (root mean square error) of 13.22 m (95th percentile). In comparison to the second EKF, slightly better results were obtained in the case without the GRU aid for IMU, and worst results with the GRU aid, with an equivalent of 3.33 m (95th percentile). The master filter (based on GRU) and fusing data from the EKFs demonstrate the best performance, with a horizontal RMSE of 0.28 m (95th percentile). Slightly worst performances were obtained for Toulouse, as can be seen in Table 2 without the INS GRU aid, due to the VO degradation in the urban environment. It can also be observed that using the Spirent GSS7000 simulator along with SimSENSOR and OKTAL-SE Sim3D software, higher errors are introduced into the simulation. Figure 4 shows a comparison between ground truth and GNSS position in NED coordinates gathered from AirSim and, respectively, OKTAL-SE Sim3D aid, affected by multipath. Although GNSS data from AirSim are not affected much by the urban canyon or the urban environment effects, data from OKTAL-SE Sim3D are more degraded, representing a more realistic positioning output.

Table 2. Positioning performance for San Francisco and Toulouse.

San Francisco			
Position source	RMSE N, (m)	RMSE E, (m)	Horizontal RMSE (95th percentile) (m)
EKF1 IMU/GNSS (no GRU aid)	5.47	2.45	13.22
EKF2 IMU/VO (no GRU aid)	3.88	2.24	10.09
Master filter (no GRU aid)	5.12	2.40	11.10
EKF1 IMU/GNSS (with GRU aid)	0.16	0.11	0.31
EKF2 IMU/ VO (with GRU aid)	1.60	1.23	3.33
Master filter	0.15	0.10	0.28
Number of VO features (95th percentile)		22,726	
Toulouse			
Position source	RMSE N (m)	RMSE E (m)	Horizontal RMSE (95th percentile) (m)
EKF1 IMU/GNSS (no GRU aid)	4.39	2.69	12.43
EKF2 IMU/VO (no GRU aid)	4.37	2.68	12.40
Master filter (no GRU aid)	0.84	3.05	7.03
EKF1 IMU /GNSS (with GRU aid)	0.12	0.09	0.31
EKF2 IMU/VO (with GRU aid)	1.15	1.10	2.42
Master filter	0.12	0.09	0.28
Number of VO features (95th percentile)		17,941	

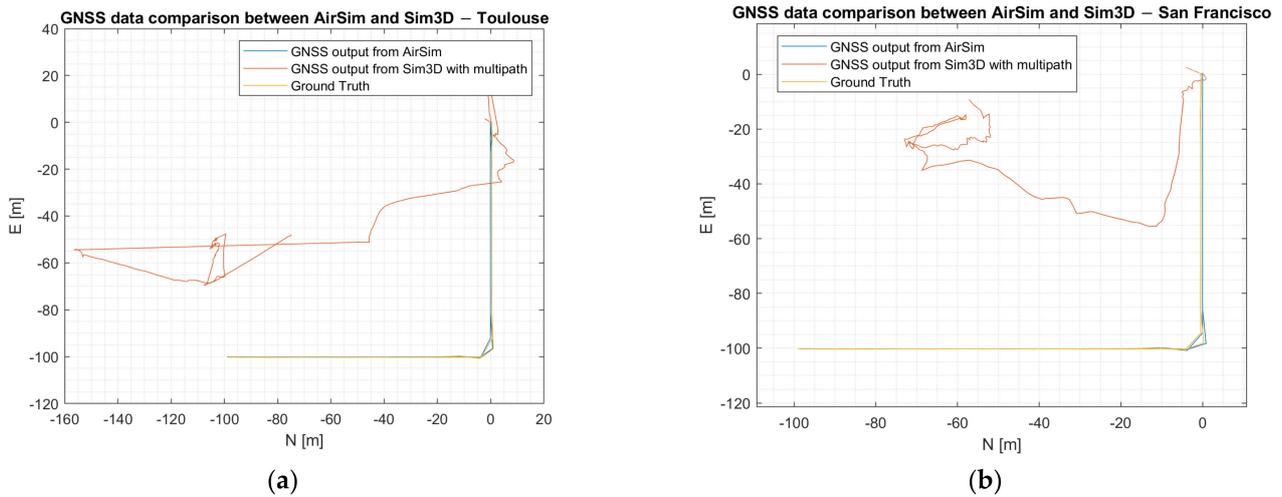


Figure 4. GNSS data with multipath in Toulouse (a) and in San Francisco (b) scenarios.

Comparing the Horizontal Dilution of Precision (HDOP) and Position Dilution of Precision (PDOP) between the Toulouse scenario and the San Francisco scenario (Figure 5), an increase in both HDOP and PDOP measurements can be observed. PDOP considers the geometry of the satellite in view and the position of the GNSS receiver. Therefore, a higher PDOP value indicates poor geometry, where the satellites are clustered in a small area of the sky, making it more difficult to determine a precise position. This is to be expected in an urban canyon scenario, where large buildings obscure direct LOS signals and reduce the area signals that can be received by the receiver. This is also visually correlated in Figure 6, where large urban canyons are present in San Francisco compared to Toulouse.

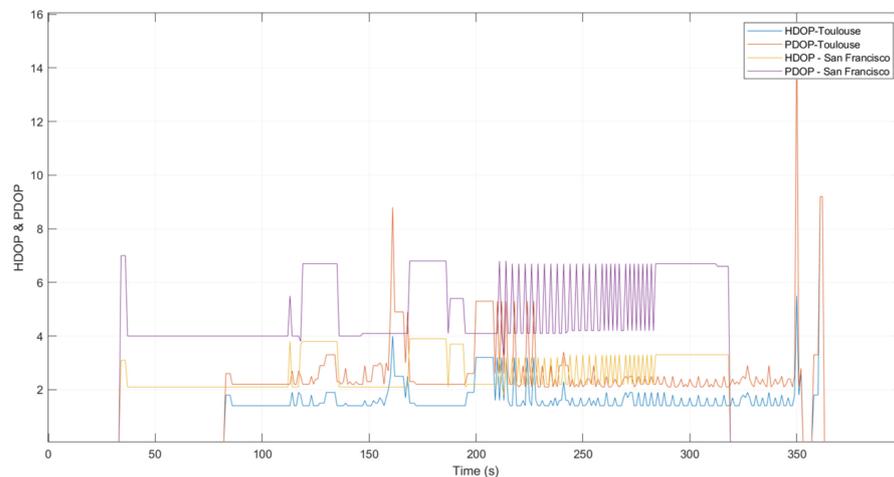


Figure 5. HDOP and PDOP for Toulouse and San Francisco scenario.

HDOP, on the other hand, specifically measures the accuracy of the horizontal position. A higher HDOP value indicates that the GPS receiver’s horizontal position calculation is less accurate, meaning there is more uncertainty in the latitude and longitude coordinates. A reduced position performance is observed in San Francisco, which represents the urban canyon scenario. This reduced performance is highlighted by the HDOP in Figure 5, which shows a higher value than seen in Toulouse. Hence, taking into consideration both HDOP and PDOP evaluation, the simulations are sensitive with regard to the change in environment and therefore aid in providing realism regarding the effects of urban canyon obscuration of LOS signals and multipath effects.

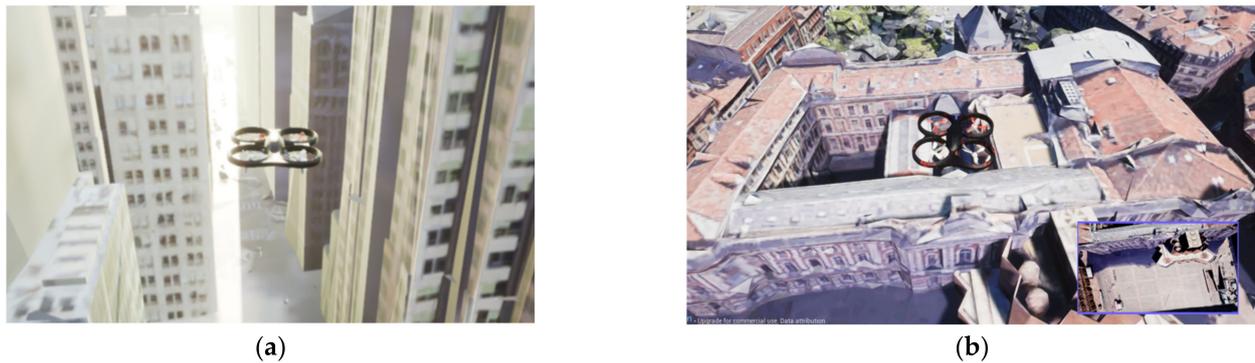


Figure 6. Feature variation of the VO algorithm in San Francisco (a) and Toulouse (b).

Considering that the UAV's monocular camera is pointing downwards during the entire mission, the number of features of the environment on the ground directly affects the VO performance. The altitude of the UAV and the light intensity are two other factors that contribute to a good VO positioning output, increasing the realism of the simulation. It can be seen clearly from Table 2 that more features were extracted when the drone was flying in an urban canyon, in San Francisco, compared to the flight in the urban environment, in Toulouse. The number of features directly affects the positioning performance of the UAV, as can be seen, for the second EKF without the GRU aid for INS, leading the San Francisco flight to a horizontal RMSE of 10.09 (95th percentile) and a horizontal RMSE of 12.40 (95th percentile) for Toulouse. As a result, it can be seen that the VO algorithm is more stable in an urban canyon, rather than an urban environment, due to building height, which leads to shades induced by the lighting effects generated by the photorealistic environment. Thus, the VO algorithm does have better performances in San Francisco, rather than in Toulouse, leading to a more accurate and stable output.

5. Conclusions

The discussed multi-sensor PNT testing, and simulation framework aims to support the development of navigation solutions for aerial and terrestrial vehicles operating in diverse and complex environments, such as urban environments. The case study based on the robust federated fusion architecture illustrates how the enhanced accuracy, resilience, and integrity of the techniques, relying on the large amounts of diverse training and testing data, can be developed, and tested ensuring repeatability and realism. Using Spirent's GSS7000, SimSENSOR, and OKTAL-SE Sim 3D in combination with AirSim, it was possible to increase the HIL simulation realism, taking into consideration stochastic and deterministic errors for the IMU, multipath, and signal obstruction for the GNSS receiver. Furthermore, the photorealistic environment, with the aid of photogrammetry data, added to the realism of the vision data simulation, being able to test and validate the accuracy of the VO algorithm under realistic conditions. With the HIL simulation set-up presented in this study, it was possible to develop, optimize and evaluate more accurately the performance of a novel sensor fusion framework, designed to be used by UAVs in urban environments.

Author Contributions: Conceptualization, R.G., G.B. and I.P.; methodology, S.A.N. and P.G.; software, S.A.N. and P.G.; validation, S.A.N. and P.G.; formal analysis, S.A.N. and P.G.; investigation, S.A.N. and P.G.; resources, S.A.N., P.G. and R.G.; data curation, S.A.N. and P.G.; writing—original draft preparation, S.A.N. and P.G.; writing—review and editing, I.P., R.G. and G.B.; visualization, S.A.N. and P.G.; supervision, I.P.; project administration, I.P. and R.G.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors from both Cranfield University and Spirent Communications PLC declare no conflict of interest.

References

1. Geragersian, P.; Petrunin, I.; Guo, W.; Grech, R. Multipath Detection from GNSS Observables Using Gated Recurrent Unit. In Proceedings of the 2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC), Portsmouth, VA, USA, 18–22 September 2022. [[CrossRef](#)]
2. Negru, S.A.; Geragersian, P.; Petrunin, I.; Zolotas, A.; Grech, R. GNSS/INS/VO fusion using Gated Recurrent Unit in GNSS denied environments. In Proceedings of the AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2023, National Harbor, MD, USA, Online, 23–27 January 2023. [[CrossRef](#)]
3. Groves, P.D.; Jiang, Z.; Rudi, M.; Strode, P. A Portfolio Approach to NLOS and Multipath Mitigation in Dense Urban Areas. In Proceedings of the 26th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2013), Nashville, TN, USA, 16–20 September 2013; pp. 3231–3247.
4. Boroujerdian, B.; Genc, H.; Krishnan, S.; Duisterhof, B.; Plancher, B.; Mansoorshahi, K.; Almeida, M.; Cui, W.; Faust, A.; Reddi, V. The Role of Compute in Autonomous Aerial Vehicles. *arXiv* **2019**, arXiv:1906.10513.
5. Giovagnola, J.; Megias, J.B.M.; Fernandez, M.M.; Cuellar, M.P.; Santos, D.P.M. AirLoop: A Simulation Framework for Testing of UAV Services. *IEEE Access* **2023**, *11*, 23309–23325. [[CrossRef](#)]
6. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. *arXiv* **2017**, arXiv:1705.05065.
7. Madaan, R.; Gyde, N.; Vemprala, S.; Brown, M.; Nagami, K.; Cristofalo, E.; Scaramuzza, D.; Schwager, M.; Kapoor, A. AirSim Drone Racing Lab. *arXiv* **2020**, arXiv:2003.05654.
8. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 564–571. [[CrossRef](#)]
9. Lawrence, P.J.; Berarducci, M.P. Comparison of Federated and Centralized Kalman Filters with Fault Detection Considerations. In Proceedings of the 1994 IEEE Position, Location and Navigation Symposium—PLANS'94, Las Vegas, NV, USA, 11–15 April 1994; pp. 703–710. [[CrossRef](#)]
10. Geragersian, P.; Petrunin, I.; Guo, W.; Grech, R. An INS/GNSS fusion architecture in GNSS denied environments using gated recurrent units. In Proceedings of the AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022, San Diego, CA, USA, Virtual, 3–7 January 2022. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.