Drone Localization using Global Navigation Satellite System and Separated Feature Visual Odometry Data Fusion

Riza Agung Firmansyah
Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Indonesia
rizaagungf@itats.ac.id (corresponding author)

Syahri Muharom
Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Indonesia
syahrimuharom@itats.ac.id

Ilmiatul Masfufiah
Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Indonesia
i.masfufiah@itats.ac.id

Ardylan Heri Kisyarangga
Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Indonesia
rangga.gonggex95@gmail.com

Dzichril Fahimatulloh Mandhia Al Farizi Rosyad
Department of Electrical Engineering, Institut Teknologi Adhi Tama Surabaya, Indonesia
dzichrilfahimatulloh@gmail.com

Received: 29 September 2024 | Revised: 22 October 2024 | Accepted: 23 November 2024

 DOI: https://doi.org/10.48084/etasr.9130

ABSTRACT

The localization system is the most important part of the overall drone navigation system. The Global Positioning System (GPS) or Global Navigation Satellite System (GNSS) is the main device commonly used in a drone. However, under certain conditions, GPS or GNSS may not function optimally, such as in situations of signal jamming or enclosed environments. This paper implemented a new approach to address this issue by combining GNSS data with Visual Odometry (VO) through Machine Learning (ML) methods. The followed process consists of three main stages. First, performing speed and orientation estimation using VO. Second, performing left and right feature separation on the images to generate a more stable and robust estimation of speed and rotation. Third, refining speed and orientation estimation by integrating GNSS data through ML-based data fusion. The proposed method strives to enhance drone localization accuracy, despite disruptions or unavailability of GNSS signals. The research results indicate that the introduced method significantly reduces Absolute Translation Error (ATE) compared to utilizing VO or GNSS separately. The average ATE produced reached 4.38 m and an orientation of 8.26°, indicating that this data fusion approach provides a significant improvement in drone localization accuracy, making it reliable in operational scenarios with limited GNSS signals.

Keywords-drone localization; data fusion; machine learning; visual odometry; GNSS

I. INTRODUCTION

The use of the GPS for global localization has been implemented in many devices. In drone technology, GPS is widely utilized for localization/,such as drones [1-3]. However, GPS has several drawbacks, involving slow signal recovery, limited accuracy, and sensitivity to interference from structures and vegetation [4. 5]. As a solution, the GNSS is considered an alternative choice because it uses more satellites, allowing for faster signal recovery and addressing potential disruptions. Thus, GNSS is well suited for use as a global localization technique. However, both GPS and GNSS have an accuracy range of up to 3 meters [6]. It is, hence, deemed necessary to improve its accuracy. To accomplish this, GNSS acting as a

coarse search can be combined with odometry techniques for a finer search [7, 8]. Odometry, a technique for tracking changes in relative position, contributes significantly to accuracy improvement [9].

VO uses cameras to track the relative motion of drones by analyzing changes in the positions of the surrounding objects, which allows for accurate pose estimation, but is susceptible to changes in lighting [10, 11]. Inertial Navigation Systems (INS) utilize inertial sensors to track the drone's movement without relying on the visual environment, but are prone to cumulative drift [12]. LiDAR-based odometry provides high accuracy and resilience to changes in lighting although it is vulnerable to weather limitations [13]. Thermal image-based odometry offers resilience to lighting changes and the ability to detect unseen objects, but it is also susceptible to decreased accuracy in certain weather conditions [14]. Meanwhile, depth odometry deploys depth sensors to estimate the distance between the drone and surrounding objects, offering the ability to operate in various lighting conditions and sensitivity to details [15, 16] although it can become inaccurate in conditions with strong light reflections. To achieve better results, some researchers are trying combinations of more than one sensor. The combination of multiple sensors has also been done before, including the use of visual inertial [17], LiDAR inertial [18], and depth inertial [19]. To obtain pose results from two or more sensors, data fusion is generally performed using the Kalman filter and its developments, such as the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF). AI-based data fusion has also been developed and shows good results. The use of Convolutional Neural Networks (CNN) [20, 21] and Long Short-Term Memory (LSTM) exhibits better results than conventional filters [22]. This paper combines GNSS data with VO to achieve more accurate results. The contributions of this paper are:

- An increase in localization accuracy when applying VO alone by separating left and right features. This technique increases the overall accuracy compared with the conventional VO [10].
- Stability in localization when GPS/GNSS signals are lost because VO can operate independently.

II. METHOD

The drone localization system depicted in Figure 1 is designed to combine GNSS data and VO to enhance the accuracy of position and speed estimation. When the drone is flown, GNSS data and camera images are recorded for offline analysis. Oriented FAST and Rotation BRIEF (ORB)-based VO and optical flow are used to accurately estimate linear movement; however, difficulties arise in estimating rotation, especially during rapid maneuvers. To address this issue, rotation estimation is calculated by comparing the average positional changes of features on the left and right sides of the image. Subsequently, the VO data are compared with GNSS through data fusion utilizing CNN, which receive inputs of speed, drone rotation, and absolute GNSS position. The CNN output, consisting of speed and rotation estimates, is used to calculate the final position of the drone deploying kinematic equations.

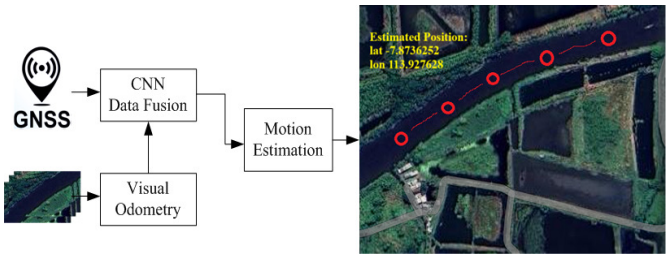

Fig. 1. Block diagram of the proposed method.

A. Experimental Approach

The aim of this experiment is to develop a more accurate drone localization system by combining data from GNSS and VO. The system is also designed to provide a backup mechanism that remains operational when GNSS signals are unavailable, such as when the drone is flying in environments with signal obstructions or interference. As a result, this system can provide more precise position estimates in various environmental conditions.

The experimental setup portrayed in Figure 2 involves the use of the FIMI X8 drone equipped with an onboard camera and GNSS as the primary reference. Additionally, the drone carries an external GNSS and an external camera as the main data collection tools. The data collected include the absolute position from GNSS as well as the images captured by the external camera, which are then utilized for VO.

Fig. 2. Experiment setup.

The experiment was conducted in three different environments: river, beach, and mountain (Figure 3). The data collection was conducted from a height of between 50 and 100 meters, using the orthogonal imaging method. This varied environmental condition aims to test the localization system's capabilities in challenging conditions, including terrain variations and potential GNSS signal interference. The drone is flown manually using a remote control within a maximum radius of 1 km from the starting point. The flights were carried out under various scenarios to explore how the system responds to changes in altitude, speed, and drone rotation. The evaluation is performed by measuring several performance metrics: position estimation accuracy in meters, speed error in meters per second, and heading error in degrees for rotation estimation. This metric provides an overview of how well the localization system performs under various flight scenarios and environments.

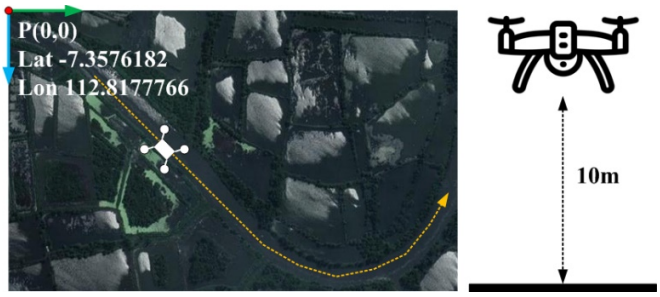

Fig. 3. Experiment location.

B. Feature Detection and Tracking

The initial stage begins with the processing of the first frame, denoted as frame t. At this stage, the algorithm performs feature extraction using ORB. ORB is a combination of the FAST feature detector and the modified BRIEF descriptor to enhance performance. ORB uses FAST to detect keypoints, then filters the best keypoints using the Harris measure, and applies a pyramid structure to detect features at various scales. To ensure feature orientation, ORB calculates the intensity-weighted centroid around the keypoint and uses the vector from the keypoint to the centroid as the orientation, making the feature more robust to rotation.

After the ORB feature is detected, each ORB feature in the image is labeled as $f_n(x, y, i, t)$, where x and y represent the position of the feature in the image, i is the sequential label number of the feature, and t is the time. Next, the feature in tracking uses optical flow. Optical flow describes the movement of each pixel between two consecutive frames in an image. Assuming that pixel intensity remains constant during the transition between frames, the basic equation for optical flow can be derived as [10]:

$$f_n(x, y, i, t) = f_n(x + u, y + v, i, t - 1) \quad (1)$$

Figure 4 shows that the same features are consistently detected in two or more consecutive frames.

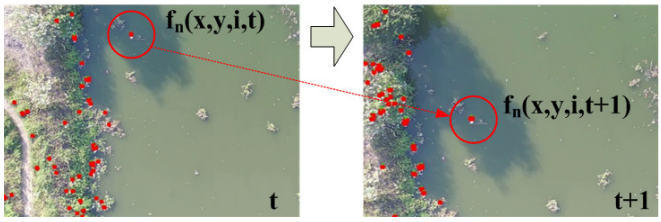

Fig. 4. Frame by frame ORB tracked feature.

C. Basic VO

The VO algorithm developed begins by reading the actual location using GNSS data. These data serve as an initial reference for the estimation of the drone's position and movement in the observed environment. After obtaining the initial location data, the video recorded by the drone is processed, where each frame captured will be used to estimate movement based on changes in the position of visual features between frame t and $t + 1$.

In frame $t + 1$, the position of each detected feature is measured again, and then the features are divided into two groups: left features and right features. This division is based on the position of the features relative to the midpoint of the image, where features located on the left of the midpoint are grouped as left features, while those on the right are categorized as right features.

The next step is to estimate the speed of the drone. This estimation is done by calculating the change in position of the features in frame $t + 1$, relative to their position in frame t. The speed estimation is calculated separately for the left feature and the right feature, and can be expressed in the form of (2) and (3). The number of features available in the image affects the movement estimation, so the appropriate number of frames will be further tested:

$$v_r = \frac{\sum_{i=1}^{n_r}(f_r(i,t) - f_r(i,t-1))}{n_r} S \quad (2)$$

$$v_l = \frac{\sum_{i=1}^{n_l}(f_l(i,t) - f_l(i,t-1))}{n_l} S \quad (3)$$

where v_r and v_l are speed estimates based on the difference in feature changes on the right and left sides, n_r and n_l are the number of detected features, f_r and f_l are the positions of the features in the image expressed in Cartesian coordinates on the x-axis and y-axis, t represents the current frame time, and S is the scaling factor that indicates the conversion from pixels to distance. After obtaining the speed estimates for both feature groups, the average total speed of the drone was calculated as:

$$v = \frac{v_r + v_l}{2} \quad (4)$$

where v is the estimated speed of the drone, v_r is the estimated speed of the right feature, and v_l is the estimated speed of the left feature. This equation provides an estimate of the total speed of the drone based on the contributions of the features on both sides of the image. Furthermore, the estimation of the drone's rotation is carried out by calculating the difference in speed between the left feature and the right feature. This estimation is crucial for determining the change in the drone's orientation between two frames. Equation (5) is used to estimate the rotation:

$$\theta = \frac{v_r - v_l}{2(F_r(x) - F_l(x))} \quad (5)$$

where θ is the estimated orientation of the drone, v_r is the estimated speed of the right feature, v_l is the estimated speed of the left feature, $F_r(x)$ is the average position of the right feature based on the x-axis, and $F_l(x)$ is the average position of the left feature based on the x-axis. Based on the estimated speed and rotation, the algorithm then calculates the new position of the drone using:

$$\Phi(t) = \Phi(0) + (\theta(t) - \theta(t-1)) \quad (6)$$

$$\mathrm{P(x,t)} = \mathrm{P(x,t-1)} + \frac{v(t)\cos\Phi(t)}{frame\ rate} \quad (7)$$

$$\mathrm{P(y,t)} = \mathrm{P(y,t-1)} + \frac{v(t)\sin\Phi(t)}{frame\ rate} \quad (8)$$

where Φ is the absolute orientation, with $\Phi(0)$ being the initial orientation, P is the position estimate, with arguments x and y

acting as the coordinate axes, t is the time, and the frame rate in this experiment is 30 FPS. This new position $P(x, y)$ must be then converted to a latitude and longitude format to be represented in geospatial coordinates. The conversion from the x-axis and y-axis coordinates to latitude and longitude is carried out using :

$$lat(t) = lat(0) + \frac{P(y,t)}{111320} \quad (9)$$

$$lon(t) = lon(0) + \frac{P(x,t)}{111320\ Cos(lat(0))} \quad (10)$$

where lat is the latitude, lon is the longitude, and the constant 111320 is the distance of one degree of latitude or longitude in meters around the equator. This constant is used because the experiment was conducted in the area around the equator.

D. Data Fusion using CNN

Data fusion is applied by integrating historical data from three main sources, namely speed (v), orientation (θ), and position data from GNSS. Speed is calculated using (4), and orientation is obtained using (5). CNN are used as a key component in this process, where the model is trained to learn patterns from the previous 10 data points to predict speed and orientation at the present time. The CNN process each input through several convolutional layers, extracting important features from the time series data. These features are then further processed to produce the final estimates of speed and orientation. The created CNN structure consists of three convolutional layers followed by a pooling layer that reduces the data dimensions, and a fully connected layer to unify all the learned features. In the final layer, the sigmoid activation function is used for the estimation of speed and rotation.

To train the CNN, the data are divided into two parts: training data and validation data. The training data are used to train the model by learning the patterns of the relationships between the historical estimates of v, θ, and GNSS. The test data are employed to evaluate the model's performance after training. This process is carried out deploying the cross-validation method to ensure that the model is evaluated properly. During training, CNN minimize prediction errors using the Mean Squared Error (MSE) loss function so that the output generated is closer to the actual value. The ration of training data to validation data is 80% to 20%. After estimating speed and orientation using CNN, the VO (4) and (5) are modified into the new data v and θ. Next, to estimate position and convert geospatial coordinates, we continue to use (6)-(10).

III. RESULTS AND DISCUSSION

In this section, testing is conducted to evaluate the proposed method. The first test aims to analyze the correlation between the number of features used (n) and the computation time, Absolute Translation Error (ATE), and orientation error. The number of features is varied to observe its effect on the processing speed as well as the accuracy of position and orientation estimation. The second test evaluates the results of data fusion using CNN, which are compared with the GNSS-only method and VO, to determine whether this data fusion can improve the accuracy of position and orientation estimation compared to other methods.

A. VO Localization

This experiment was conducted to determine the correlation between the number of features extracted in the VO process and three important variables: computation time, ATE, and orientation error. The aim of this experiment is to understand how variations in the number of features used to track the movement of objects in images affect the accuracy of position and orientation estimation, while also considering computation speed. This experiment was conducted by varying the number of features used in the VO process, from 50 to 500 features in increments of 50 features.

Table I illustrates the results of the experiments that examined computation time, ATE, and orientation errors in the VO system. These results are very important. First, regarding computation time, it was found that the time required to process data is relatively constant, ranging from 0.022 to 0.028 seconds, regardless of the number of the features utilized. This can be explained by the ORB feature detection mechanism, which is a key factor in determining computation time. Computation time tends to remain stable when there is no significant feature loss, as the system only re-detects ORB features when there is feature loss. Therefore, as long as the system successfully maintains features without a significant loss, the computation time remains consistent, even as the number of features increases.

TABLE I. SEPARATED ORB FEATURE VO RESULT

Features (n)	**Metrics**		
	Computation time (s)	**ATE (m)**	**Orientation Error (°)**
50	0.028	16.19	18.97
100	0.023	11.83	17.88
150	0.024	11.45	16.52
200	0.024	12.08	16.16
250	0.022	11.76	15.41
300	0.023	14.41	14.21
350	0.023	**10.75**	**14.12**
400	0.024	12.54	15.34
450	0.023	11.55	16.93
500	0.027	11.20	16.43

Next, the ATE metric shows a clear downward trend as the number of features increases from 50 to 350 features. At 50 features, the recorded ATE was 16.193 meters, which is the highest value compared to the number of other features. The performed analysis shows that having too few features, such as 50, increases the risk of feature loss, which is the loss of features tracked by the algorithm, thereby significantly reducing positional accuracy. With a small number of features, the system does not have enough information to accurately track movements, resulting in a large ATE. However, as the number of features increases, reaching above 100, the ATE begins to decrease and tends to stabilize. This is because feature loss becomes less frequent with a larger number of features, allowing the system to maintain stable accuracy. At 350 features, the ATE reached a low point of 10.750 meters although at larger feature counts the ATE experienced slight fluctuations without significant changes.

In terms of orientation errors, the conducted experiments show a different pattern. Orientation errors are generally not

directly affected by the total number of features, but rather by the distribution of features between the left and right sides of the image. When there is an imbalance in the number of features detected on both sides of the image, orientation errors tend to increase. For example, at 350 features, the orientation error rises sharply to 28.11941°, indicating a significant imbalance between features on the left and right sides. This suggests that orientation errors are more related to the imbalance in feature distribution than to the total number of features used. After 350 features, the orientation error decreases again and stabilizes around 15° to 16°, indicating that the feature distribution becomes more balanced, even as the number of features continues to increase.

B. GNSS and VO Fusion

Based on the VO estimation results, the drone's path deviates significantly compared to the Ground Truth path, as reflected by an ATE of 10.75 m. This error is most likely caused by the accumulation of drift error that often occurs in VO, especially in complex and winding environments. In addition, the orientation estimation by VO demonstrates an error of 14.12°, indicating a lack of precision in calculating the change in direction or rotation of the drone.

The accuracy improved significantly after the data fusion between VO and GNSS. The ATE dropped to 4.38 m. This indicates that GNSS data can provide a more stable and accurate position reference, thereby correcting the cumulative errors from VO. Furthermore, the orientation error has been reduced to 8.26°. Table II displays the comparison results between the VO method alone and the fusion method. With the availability of GNSS data, the navigation system is able to reduce orientation errors that were previously caused by the limitations of VO in tracking angular changes with precision.

TABLE II. OVERALL RESULT COMPARISON

Method	Metrics		
	Computation time (s)	ATE (m)	Orientation Error (°)
VO	0.023	10.75	14.12
VO+GNSS fusion	0.03	4.38	8.26

In terms of computation time, VO alone takes 0.023 seconds, which is slightly faster than VO+GNSS fusion, which takes 0.03 seconds. Although the computation time has slightly increased, this is considered reasonable given the additional process of integrating data from GNSS. This small increase in computation time is still acceptable in real-time applications, and the significant improvement in accuracy makes the trade-off in computation time very worthwhile. Overall, these results indicate that the use of VO+GNSS fusion significantly improves the performance of drone navigation systems in terms of position and orientation accuracy. Figure 5 depicts the drone's trajectory when using 350 features. This trajectory is plotted deploying GPS visualizer based on the data that have been obtained.

Fig. 5. Trajectory comparison result.

IV. CONCLUSIONS

On the basis of the obtained data, the Visual Odometry (VO) method independently exhibits a high error rate, with an Absolute Translational Error (ATE) of 10.75 meters and an orientation error reaching 14.12 degrees. This reflects a significant deviation between the estimates and the actual position, as well as a lack of accuracy in the directional calculations. However, after applying VO and Global Navigation Satellite System (GNSS) fusion, there was a significant improvement in performance compared to conventional VO [10]. The ATE decreased to 4.38 m, and the orientation error was reduced to 8.26 °, indicating a tangible improvement in position and orientation estimation. The main advantage of VO+GNSS fusion is its reliability, offering good localization even in the case of sensor collapse, with VO compensating during GNSS weakness and vice versa. In the future, the integration of thermal imaging will enable VO to operate in low-light conditions, improving performance in scenarios where lighting is a challenge.

ACKNOWLEDGMENT

The authors would like to express their deepest gratitude to the Directorate of Technology Research and Community Service (DRTPM) of the Ministry of Education and Culture of the Republic of Indonesia for the Penelitian Fundamental Reguler (PFR) grant scheme in 2024 with Master contract number 109/E5/PG.02.00.PL/2024 dated June 11, 2024, derivative contracts 083/SP2H/PT/LL7/2024 dated June 12, 2024, and 08/KP/LPPM/ITATS/2024 dated June 13, 2024.

REFERENCES

[1] B. Van den Bergh and S. Pollin, "Keeping UAVs Under Control During GPS Jamming," *IEEE Systems Journal*, vol. 13, no. 2, pp. 2010–2021, Jun. 2019, https://doi.org/10.1109/JSYST.2018.2882769.

[2] F. Jametoni and D. E. Saputra, "A Study on Autonomous Drone Positioning Method," in *2021 Sixth International Conference on Informatics and Computing*, Jakarta, Indonesia, 2021, pp. 1–5, https://doi.org/10.1109/ICIC54025.2021.9632926.

[3] F. Alotaibi, A. Al-Dhaqm, and Y. D. Al-Otaibi, "A Conceptual Digital Forensic Investigation Model Applicable to the Drone Forensics Field," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11608–11615, Oct. 2023, https://doi.org/10.48084/etasr.6195.

[4] S. Mulugeta and T. Kassa, "Investigation of GPS Loss of Lock Occurrence and its Characteristics Over Ethiopia using Geodetic GPS

Receivers of the IGS Network," *Advances in Space Research*, vol. 69, no. 2, pp. 939–950, Jan. 2022, https://doi.org/10.1016/j.asr.2021.10.035.

[5] A. Hussain, A. Ahmed, H. Magsi, J. B. Soomro, S. S. H. Bukhari, and J.-S. Ro, "Adaptive Data Length Method for GPS Signal Acquisition in Weak to Strong Fading Conditions," *Electronics*, vol. 10, no. 14, Jul. 2021, Art. no. 1735, https://doi.org/10.3390/electronics10141735.

[6] W. Magiera *et al.*, "Accuracy of Code GNSS Receivers under Various Conditions," *Remote Sensing*, vol. 14, no. 11, May 2022, Art. no. 2615, https://doi.org/10.3390/rs14112615.

[7] R. A. Firmansyah, W. S. Pambudi, T. Suheta, E. A. Zuliari, S. Muharom, and M. B. S. Hidayatullah, "Implementation of Artificial Neural Networks for Localization System on Rescue Robot," in *2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar*, Batu, Indonesia, 2018, pp. 305–309, https://doi.org/10.1109/EECCIS.2018.8692861.

[8] R. A. Firmansyah, T. A. Sardjono, and R. Mardiyanto, "Improving the Adaptive Monte Carlo Localization Accuracy using a Convolutional Neural Network," *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, vol. 12, no. 3, pp. 167–174, Aug. 2023, https://doi.org/10.22146/jnteti.v12i3.7432.

[9] Y. Chen, B. Xu, B. Wang, J. Na, and P. Yang, "GNSS Reconstrainted Visual–Inertial Odometry System using Factor Graphs," *IEEE Geoscience and Remote Sensing Letters*, vol. 20, pp. 1–5, 2023, https://doi.org/10.1109/LGRS.2023.3236803.

[10] H. Bavle, P. De La Puente, J. P. How, and P. Campoy, "VPS-SLAM: Visual Planar Semantic SLAM for Aerial Robotic Systems," *IEEE Access*, vol. 8, pp. 60704–60718, 2020, https://doi.org/10.1109/ACCESS.2020.2983121.

[11] T. Zhang, C. Liu, J. Li, M. Pang, and M. Wang, "A New Visual Inertial Simultaneous Localization and Mapping (SLAM) Algorithm Based on Point and Line Features," *Drones*, vol. 6, no. 1, Jan. 2022, Art. no. 23, https://doi.org/10.3390/drones6010023.

[12] E. Petritoli, F. Leccese, and G. S. Spagnolo, "Inertial Navigation Systems (INS) for Drones: Position Errors Model," in *2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, 2020, pp. 500–504, https://doi.org/10.1109/MetroAeroSpace48742.2020.9160304.

[13] M. Gazzea, M. Pacevicius, D. O. Dammann, A. Sapronova, T. M. Lunde, and R. Arghandeh, "Automated Power Lines Vegetation Monitoring using High-Resolution Satellite Imagery," *IEEE Transactions on Power Delivery*, vol. 37, no. 1, pp. 308–316, Feb. 2022, https://doi.org/10.1109/TPWRD.2021.3059307.

[14] Y. Wang, H. Chen, Y. Liu, and S. Zhang, "Edge-Based Monocular Thermal-Inertial Odometry in Visually Degraded Environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 4, pp. 2078–2085, Apr. 2023, https://doi.org/10.1109/LRA.2023.3246381.

[15] S. Muharom, R. A. Firmansyah, and Y. A. Prabowo, "Local Generating Map System using Rviz ROS and Kinect Camera for Rescue Robot Application," *International Journal of Electronics and Telecommunication*, vol. 69, no. 3, pp. 621–626, Jul. 2023.

[16] J. Liu, X. Li, Y. Liu, and H. Chen, "RGB-D Inertial Odometry for a Resource-Restricted Robot in Dynamic Environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9573–9580, Oct. 2022, https://doi.org/10.1109/LRA.2022.3191193.

[17] G. Fink, M. Franke, A. F. Lynch, K. Röbenack, and B. Godbolt, "Visual Inertial SLAM: Application to Unmanned Aerial Vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 1965–1970, Jul. 2017, https://doi.org/10.1016/j.ifacol.2017.08.162.

[18] H. Tang, X. Niu, T. Zhang, L. Wang, and J. Liu, "LE-VINS: A Robust Solid-State-LiDAR-Enhanced Visual-Inertial Navigation System for Low-Speed Robots," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, 2023, https://doi.org/10.1109/TIM.2023.3260279.

[19] P. Gu and Z. Meng, "S-VIO: Exploiting Structural Constraints for RGB-D Visual Inertial Odometry," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3542–3549, Jun. 2023, https://doi.org/10.1109/LRA.2023.3270033.

[20] L. Crupi, E. Cereda, A. Giusti, and D. Palossi, "Sim-to-Real Vision-Depth Fusion CNNs for Robust Pose Estimation Aboard Autonomous Nano-quadcopters," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Detroit, MI, USA, Oct. 2023, pp. 7711–7717, https://doi.org/10.1109/IROS55552.2023.10342162.

[21] H. Jo and E. Kim, "New Monte Carlo Localization using Deep Initialization: A Three-Dimensional LiDAR and a Camera Fusion Approach," *IEEE Access*, vol. 8, pp. 74485–74496, 2020, https://doi.org/10.1109/ACCESS.2020.2988464.

[22] S. Guan and X. Luo, "Fusing Ultra-wideband Range Measurements with IMU for Mobile Robot Localization," in *2021 11th International Conference on Intelligent Control and Information Processing*, Dali, China, 2021, pp. 107–111, https://doi.org/10.1109/ICICIP53388.2021.9642157.