Unmanned Aerial Vehicle Localization Using Angle of Departure from a Single Base Station and Dead-Reckoning

Akash Rajasekaran,[1] Mehari Meles,[1] Reino Virrankoski,[1] and Riku Jäntti[1]
[1]*Department of Information and Communications Engineering, Aalto University School of Electrical Engineering 02150 Espoo, Finland*
Email: akash.rajasekaran@aalto.fi, mehari.meles@aalto.fi, reino.virrankoski@aalto.fi, riku.jantti@aalto.fi

Localization in GNSS-denied environments for Unmanned Aerial Vehicles (UAVs) has recently gained significant interest from the research community. Most of the research is focused primarily on visual localization. This paper, examines an algorithm which employs Angle of Departure (AoD) and UAVs payload sensor data for UAV localization. First the algorithm uses multiple AoDs from a single base station and a travel calculated by applying dead-reckoning on the UAVs Inertial Measurement Unit (IMU), to compute UAV location in two-dimensional (2D) coordinates. The 2D location estimate is then fed into a modified Extended Kalman Filter (EKF), which employs the estimate, IMU and barometer data to compute the three-dimensional (3D) coordinates for UAV. For the simulation, we applied Simulation-in-the-Loop (SITL) accompanied by Arducopter and MAVLink to simulate different trajectories and collect the required data for the algorithm. We validated our algorithm by comparing the EKF estimates with IMU dead-reckoned positions. Three simulations were performed, consisting of linear, zigzag and curved trajectories. We achieved a 90^{th} percentile error of 2.5m and 4m for the x-coordinate and y-coordinate, respectively, on the zigzag and curved trajectories. Interestingly, the linear trajectory showed a larger localization error in its y-coordinate.

Introduction: The use case of autonomous Unmanned Aerial Vehicles (UAVs) has been drastically increasing over the past few years such as for search and rescue, fire detection and mitigation and boarder security [1–3]. The UAV navigation system should be well balanced and redundant. Autonomous operation in dynamic and unknown environments require precise knowledge of UAV's state, which is characterized by position, velocity and attitude [4]. Currently, almost all commercial and defence UAVs need Global Navigation Satellite System (GNSS) with the help of Inertial Navigation System (INS) to navigate in unknown environments [5]. However, the continuous availability of satellite positioning is not guaranteed. In certain environments and situations, such as indoors, dense forest, near buildings (multipath) and in the presence of jamming or spoofing, it is unreliable [6].

To overcome these challenges, many researchers have proposed algorithms and systems based on visual sensors. Further, the scope of computer vision algorithms for path-planning, mapping and localization have been examined [7]. Absolute Visual Localization (AVL) has been introduced, adopting the reference satellite map for feature matching for localization [8]. An algorithm has been developed that fuses stereo visual odometry and inertial measurements using a Kalman filter [9].

Localization using Radio Frequency (RF) signals has garnered considerable interest among researchers, particularly regarding cellular networks due to the already established infrastructure. A survey in [10] provides a thorough analysis of RF-based localization systems adopting to various radio communication techniques and localization techniques for UAV location. The most widely adopted techniques for UAV localization are based on Angle of Departure (AoD), Angle of Arrival (AoA), Time of Arrival (ToA), Time Difference of Arrival (TDoA) and Received Signal Strength Indicator (RSSI) measurements. When these techniques are applied, usually a minimum of two base stations are required. In our previous work [11], we conducted an analysis of AoA estimation implementing two base stations with known locations transmitting RF signals.

This paper, examines the localization of an UAV using a single base station (BS) with known location that transmits RF signals and the AoD calculated at the UAV. The final estimate for the UAV 3D coordinates is computed adopting Extended Kalman Filter (EKF) to fuse Inertial Measurement Unit (IMU) data, barometer data and UAV's 2D location estimate. Since the system is configured for a single BS and relatively little research has been conducted regarding UAV localization by employing a single BS [12], we assessed our results against those achieved through IMU based dead-reckoning [13]. Simulations were performed in Simulation-in-the-Loop (SITL) software accompanied by Arducopter firmware for a quadcopter. We primarily concentrated on exploring three distinct trajectories that are commonly employed in practical settings. The simulation section provides more information on the trajectory.

Two-Dimensional Cartesian coordinate estimation using multiple AoD: This section, discusses the algorithm to estimate the 2D coordinate of the UAV using AoD angles, yaw angles and distance traveled between each AoD measurements. Figure 1, shows the 3D model of the environment with all the parameters required to calculate the 3D position of the UAV. Heights h_0 and h_1 are obtained by fusing IMU and barometer height in the EKF. Figure 2 presents the parameters used to calculate the 2D location of the UAV. The main concept of the algorithm is to use multiple azimuth angles (θ), yaw angle (ψ) and distance travelled (D) to solve a linear equation using a matrix (4). The main parameter to determine is the projected range $\hat{d}$. From which we can determine the X and Y coordinates in the NED (North, East and Down) frame by converting the polar coordinates $\hat{d}_i$ and θ_i. From Figure 2 we calculate D_j as:

$$\begin{aligned} D_1 &= -p_0 + p_1 \\ D_1 &= |D_1|e^{i\psi_1},\ p_0 = \hat{d}_0 e^{i\theta_0} \text{ and } p_1 = \hat{d}_1 e^{i\theta_1} \\ D_j &= |D_j|e^{i\psi_j}, \quad j = 1, 2 \ldots N \\ p_i &= \hat{d}_i e^{i\theta_i}, \qquad i = 0, 1 \ldots N \end{aligned} \tag{1}$$

where, D_j is the distance travelled, in other words the difference between the dead reckoned position and the previous location estimate. ψ_j is the yaw angle that is the heading of the UAV with respect to north, p_i is desired position at the i^{th} measurement of AoD, $\hat{d}_i$ is the projected range in meter, θ_i is the azimuth angle from AoD in radians and N is the number of measurements.

From (1), we can write the real and imaginary parts of the equation as:

$$\begin{aligned} -\hat{d}_0 cos(\theta_0) + \hat{d}_1 cos(\theta_1) &= |D_1| cos(\psi_1) \\ -\hat{d}_0 sin(\theta_0) + \hat{d}_1 sin(\theta_1) &= |D_1| sin(\psi_1) \\ -p_0 + p_N &= \sum_{j=1}^{N} D_j \end{aligned} \tag{2}$$

Equation (2) can be written in a matrix form by applying multiple measurements of all the parameters mentioned above as:

$$\underbrace{\begin{bmatrix} -cos(\theta_0) & cos(\theta_1) & 0 & . & 0 \\ -sin(\theta_0) & sin(\theta_1) & 0 & . & 0 \\ -cos(\theta_0) & 0 & cos(\theta_2) & . & 0 \\ -sin(\theta_0) & 0 & sin(\theta_2) & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ -cos(\theta_0) & 0 & 0 & . & cos(\theta_N) \\ -sin(\theta_0) & 0 & 0 & . & sin(\theta_N) \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \hat{d}_0 \\ \hat{d}_1 \\ \hat{d}_2 \\ . \\ . \\ \hat{d}_N \end{bmatrix}}_{\hat{d}} =$$

$$\underbrace{\begin{bmatrix} cos(\psi_1) & 0 & 0 & . & 0 \\ sin(\psi_1) & 0 & 0 & . & 0 \\ cos(\psi_1) & cos(\psi_2) & 0 & . & 0 \\ sin(\psi_1) & sin(\psi_2) & 0 & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ cos(\psi_1) & cos(\psi_2) & 0 & . & cos(\psi_N) \\ sin(\psi_1) & sin(\psi_2) & 0 & . & sin(\psi_N) \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} |D_1| \\ |D_2| \\ . \\ . \\ |D_N| \end{bmatrix}}_{D} \tag{3}$$

which can be written as:

$$\hat{d} = A^{-1} B D \tag{4}$$

$$\begin{aligned} X_{m,i} &= \hat{d}_i cos(\theta_i) \\ Y_{m,i} &= \hat{d}_i sin(\theta_i) \end{aligned} \tag{5}$$

We assume the position of the BS to be at (0,0) in the NED frame, we can calculate the location of the UAV ($X_{m,i}$,$Y_{m,i}$) from (5), which

will be applied as measurements in the update step of the EKF. The application of multiple AoDs will improve the localization result. In our case the EKF estimate improves the predicted state after the first five AoD measurements are implemented in the computation.

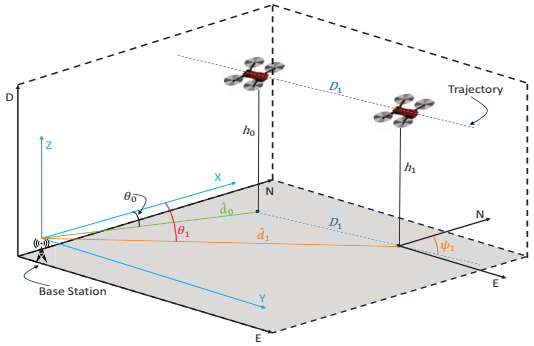

Fig 1 *3D Model setup*

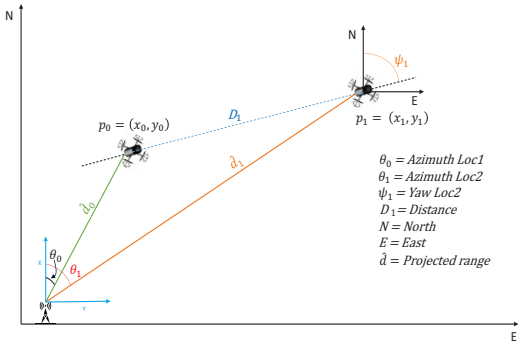

Fig 2 *2D Model setup*

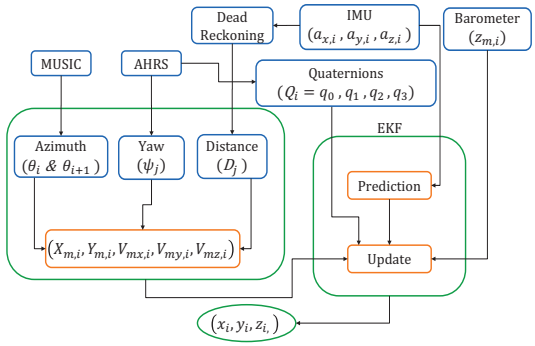

Fig 3 *Algorithm architecture*

EKF sensor Fusion: This section, analyzes the implementation of the EKF algorithm. Figure 3 shows the algorithm architecture. Here, for simplicity and to validate the algorithm instead of implementing a Multiple Signal Classification (MUSIC) algorithm, we used the spherical coordinates of the GPS with an additional noise $\theta_n \sim \mathcal{N}(0, \sigma_\theta^2)$, where $\sigma_\theta^2 = 5°$, as the measured azimuth angle θ_i. The present solution relies on an EKF due to the nonlinear nature of the process and measurements model, because the classic Kalman Filter functions properly only under the linear assumption. The EKF is utilized to compute estimates of attitude, position, and velocity based on noisy sensor readings.

Kalman filters operates a continuous loop of prediction and updating [14]. The system status at the next time frame is determined by the present states and the system inputs. In the attitude calculations, the angular rate-sensor measurements generate the input; on the other hand, accelerometer measurements serve as the basis for velocity and position calculations. The update phase rectifies the predicted states to account for discrepancies in the measurement signals (e.g., sensor bias and drift) using true attitude, position, and velocity estimates. The attitude is collected from the UAV system's AHRS (Attitude and Heading Reference System), height from the barometer, and 2D Cartesian coordinates computed by the multiple AoD algorithm. For the velocity measurement we apply the time-distance-velocity formula, we calculate the difference between the dead reckoned position and the previous location estimate to attain the distance. As for the time, AoD measurements have 0.2 seconds sampling interval, because the GPS measurements are taken on a 5 Hz frequency.

$$\begin{aligned} \vec{x}_{k|k-1} &= f(\vec{x}_{k-1|k-1}, \vec{u}_{k|k-1}) + \vec{w}_{k-1} \\ P_{k|k-1} &= F_k P_{k-1|k-1} {F_k}^T + Q_{k-1} \\ F_{k-1} &= \frac{\partial f}{\partial x}\Big|_{\vec{x}_{k-1}, \vec{u}_k} \end{aligned} \tag{6}$$

Equation group (6), presents the discrete-time prediction equations where the states propagate in time adopting the acceleration, gyroscope and magnetometer sensor readings. $\vec{x}_{k|k-1}$ is the state prediction model and $P_{k|k-1}$ is the covariance estimate. F_{k-1} and Q_{k-1} are the process jacobian and the process-noise covariance matrices, respectively. Since the state-transition model is nonlinear, the covariance cannot be propagated with a direct implementation of the state-transition vector $\vec{f}$. To address this issue, $\vec{f}$ is linearized in every EKF computation round by taking partial derivatives with respect to the system state $\vec{x}$, which generates the process jacobian F.

$$\begin{aligned} S_k &= H_k P_{k|k-1} {H_k}^T + R_k \\ K_k &= P_{k|k-1} H_k^T {S_k}^{-1} \\ \vec{x}_{k|k} &= \vec{x}_{k|k} + K_k(\vec{z}_k - \vec{h}_k) \\ P_{k|k} &= (I - K_k H_k) P_{k|k-1} (I - K_k H_k)^T + K_k R_k K_k^T \\ H_k &= \frac{\partial h}{\partial x}\Big|_{\vec{x}_k} \end{aligned} \tag{7}$$

Equation group (7), consists of the final stage of the EKF. The discrete-time update equations are used to update or correct the predicted state estimate based on the quality of the process models, process inputs and measurements. S_k, K_k, $\vec{x}_{k|k}$, $P_{k|k}$ and H_k denote the innovation covariance, kalman gain, updated state estimate, updated covariance estimate and observation jacobian, respectively. H_k is defined as the partial derivative of the observation vector $\vec{h}_k$ with respect to the state $\vec{x}_k$.

$$\vec{x} = \begin{Bmatrix} \vec{p}^N \\ \vec{v}^N \\ {}^N\vec{q}^B \\ \vec{a}^B_{bias} \\ \vec{w}^B_{bias} \end{Bmatrix} = \begin{Bmatrix} \text{NED Position}(3) \\ \text{NED Velocity}(3) \\ \text{Body Attitude}(4) \\ \text{Acceleration Bias}(3) \\ \text{Angular} - \text{Rate Bias}(3) \end{Bmatrix} \tag{8}$$

The state vector $\vec{x}_k$ has 16 states as seen in (8), namely the position $(\vec{p}^N)$, velocity $(\vec{v}^N)$, acceleration bias $(\vec{a}^B_{bias})$, and angular rate bias $(\vec{w}^B_{bias})$, each having three coordinates and finally the quaternions $({}^N\vec{q}^B)$ are used to characterize the body attitude.

$$f(\vec{x}_{k-1}, \vec{u}_{k-1}) = \begin{Bmatrix} \vec{p}^N_{k-1} + \vec{v}^N_{k-1} \cdot dt \\ \vec{v}^N_{k-1} + \left[{}^N R^B_{k-1} \cdot \left(\vec{a}^B_{meas,k} - \vec{a}^B_{bias,k} \right) - \vec{a}^N_{grav,k} \right] \cdot dt \\ \left[I_4 + \frac{dt}{2} \cdot \left(\Omega_{meas,k} - \Omega_{bias,k} \right) \right] \cdot {}^N\vec{q}^B_{k-1} \\ \vec{a}^B_{bias,k-1} \\ \vec{w}^B_{bias,k-1} \end{Bmatrix} \tag{9}$$

$$\vec{w}_{k-1} = \begin{Bmatrix} - {}^N R^B_{k-1} \cdot \vec{a}^B_{noise} \cdot dt^2 \\ - {}^N R^B_{k-1} \cdot \vec{a}^B_{noise} \cdot dt \\ - \frac{dt}{2} \cdot \Xi_{k-1} \cdot \vec{w}^B_{noise} \\ \vec{N}(0, \sigma^2_{dd,a}) \\ \vec{N}(0, \sigma^2_{dd,w}) \end{Bmatrix}, \quad \Xi_{k-1} = \begin{bmatrix} \vec{q}_v^T \\ q_0 . I_3 + [\vec{q}_v \times] \end{bmatrix} \tag{10}$$

The state-transition model presented in (9) - (10) is integral to the EKF prediction stage. It presents the equations that propagate the system states from one time-step to the next, utilizing high-quality sensors as inputs. Additionally, the model also includes the process-noise vectors linking each state to sensor noise. Moreover, they enable computing

of the process covariance matrix, Q, and process Jacobian, F; both are pivotal for propagating the system covariance, P, from one time-step to subsequent time-steps. $\vec{f}$ represents the state-transition vector with five individual models and $\vec{w}_{k-1}$ symbolize, the process noise vector shown in Equation10, where $\vec{N}$ depicts the sensor noise vector corresponding to the angular-rate and accelerometer bias with zero-mean Gaussian distribution and variance σ^2_{dd}.

Simulations: Simulations were executed through a Flight Dynamics Model (FDM) to replicate the physical components associated with UAV motion. Inputs from a SITL program, running ArduPilot firmware, are accepted and simulation outputs such as vehicle status, position and velocities are then provided back to the firmware. This mimics the behaviour of sensors within a true-world environment. Mavproxy with Pymavlink was used to communicate and collect the IMU and other telemetry data from the UAV. The telemetry data we collected from the IMU consisted of the measurements of following sensors: accelerometer, gyroscope, magnetometer at 100 Hz, barometer with absolute pressure at 10 Hz, AHRS providing roll, pitch and yaw at 10 Hz, and GPS coordinates (WGS84) at 5 Hz. Finally, we collected the output of the UAV navigation system (position and velocity at 10 Hz) for algorithm validation.

The simulations focused on trajectories that mimic real-world scenarios. A zigzag trajectory, mainly used for land survey missions, a linear trajectory, and finally a curved trajectory. The curved trajectory is analogous to zigzag and is utilized to illustrate how the new algorithm stacks up against IMU dead-reckoning over time.

Results: From the three trajectories simulated, we visualize the zigzag trajectory. Figure 4 shows the 3D trajectory of the estimated values and the data from the UAV navigation system. It can be seen that the EKF estimated trajectory and the UAV trajectory are similar with several centimetre level deviation, mean of 5cm, in the height. Since we are using a single base station and there are limited methods implemented using single base station for localization, we compared our algorithm to IMU dead-reckoning. It is known that localizing with dead-reckoning is very dependent on time due to the bias and noise accumulation.

We evaluated the accuracy of the EKF and IMU for the X and Y coordinates with respect to the UAV navigation system, as illustrated in Figures 5 and 6, which displays the Cumulative Distribution Function (CDF) for both coordinates. It is clearly visible that the EKF outperforms the IMU dead-reckoning. 90% of the error in the X-coordinate is below 2.56 m and below 4.08 m in the Y-coordinate, with both coordinates having a maximum error below 5 m. Table. 1 provides a more detailed error analysis for all the trajectories in the simulations. It includes the 90^{th} percentile, mean and the median error. The results show that the linear trajectory has larger position error especially in the y-coordinate compared to other trajectories. This is due to the calculation of the distance travelled using the IMU. For the linear trajectory, we simulated the drone to fly parallel to the y-axis, hence the y-coordinate of the accelerometer was varying largely and accumulating more error effecting the distance traveled calculation.

Conclusion: In this paper, we presented an algorithm for UAV localization and evaluated its performance through simulations. The algorithm uses single base station's AoD masurements, IMU dead-reckoning and custom EKF for location computation. Algorithm performance was evaluated by simulating it with three different UAV trajectories, which are deployed in practice in real world scenarios. The results show that for a dynamic trajectory like zigzag and a curved trajectory the algorithm is able to achieve a mean error ranging from 1m to 2m, when compared to IMU dead-reckoning, which has a mean error above 100m. For the linear trajectory, the calculation of the distance travelled effects the 2D

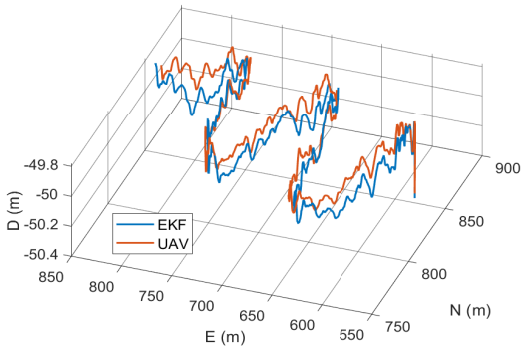

Fig 4 *3D Trajectory*

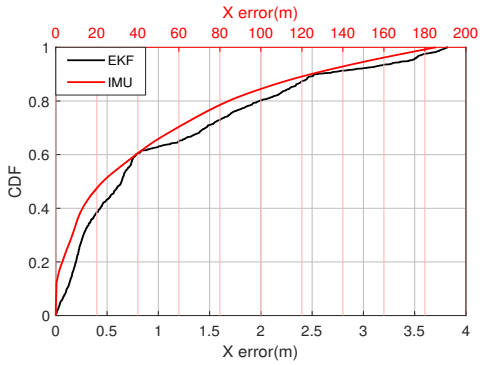

Fig 5 *X-Error CDF*

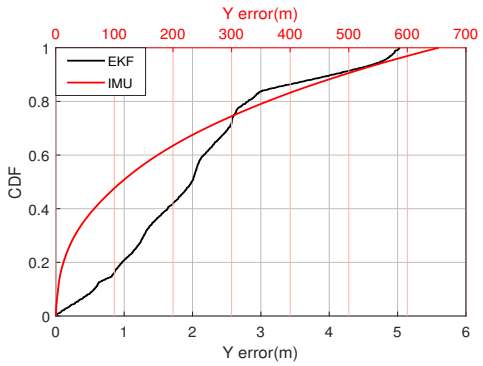

Fig 6 *Y-Error CDF*

estimates due to the accumulation of error over time. The algorithm is successful in achieving an accuracy of 1m to 5m for dynamic trajectories and 1m to 10m for linear trajectories, with the caveat that the UAV must remain in motion and AoD measurements must be taken every 0.2 seconds. Future work would involve testing the algorithm with actual data from the real world and conducting outdoor experiments.

Author contributions: **Akash Rajasekaran**: Conceptualization; Data curation; Investigation; Formal analysis; Methodology; Validation; Visualization; Writing - original draft, review and editing. **Mehari Meles**: Investigation; Methodology; Validation; Writing - review and editing. **Reino Virrankoski**: Supervision; Writing - review and editing. **Riku Jäntti**: Methodology; Supervision; Writing - review and editing.

Table 1. Error analysis

Error	90^{th} Percentile (m)						Mean (m)						Median (m)					
Trajectory	Linear		Zigzag		X-curve		Linear		Zigzag		X-curve		Linear		Zigzag		X-curve	
Method	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU	EKF	IMU
X	4.15	52.62	2.56	124.7	2.28	42.62	2.58	32.5	1.05	44.25	0.95	17.58	2.85	34.84	0.64	23.08	0.50	13.98
Y	11.72	261.6	4.08	92.65	2.74	167.54	6.47	139.3	2.05	182.5	1.36	62.17	6.36	132.43	1.99	112.46	1.26	37.94

Acknowledgments: The authors would like to acknowledge partial financial support from Business Finland project MULTICO (n:o 6598/31/2019).

Conflict of interest statement: The authors declare no conflicts of interest.

Data availability statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. Scherer, J., et al.: An autonomous multi-uav system for search and rescue. In: Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use. DroNet '15, p. 33–38. New York, NY, USA: Association for Computing Machinery (2015). https://doi.org/10.1145/2750675.2750683
2. Shakhatreh, H., et al.: Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges. IEEE Access 7, 48572–48634 (2019). doi:10.1109/ACCESS.2019.2909530
3. Azmat, M., Kummer, S.: Potential applications of unmanned ground and aerial vehicles to mitigate challenges of transport and logistics-related critical success factors in the humanitarian supply chain. Asian Journal of Sustainability and Social Responsibility 5 (2020). doi:10.1186/s41180-020-0033-7
4. Kabiri, M., et al.: A review of radio frequency based localisation for aerial and ground robots with 5g future perspectives. Sensors 23(1) (2023). doi:10.3390/s23010188. https://www.mdpi.com/1424-8220/23/1/188
5. Cheng, J., Guan, K., Quitin, F.: Virtual multiantenna array for estimating the doa of a transmitter in uav-assisted networks. In: 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, , pp. 1–6. (2020)
6. Lin, H.Y., Zhan, J.R.: GNSS-denied UAV indoor navigation with UWB incorporated visual inertial odometry. Measurement 206, 112256 (2023). doi:10.1016/j.measurement.2022.112256. https://doi.org/10.1016/j.measurement.2022.112256
7. Al-Kaff, A., et al.: Survey of computer vision algorithms and applications for unmanned aerial vehicles. Expert Systems with Applications 92 (2017). doi:10.1016/j.eswa.2017.09.033
8. Couturier, A., Akhloufi, M.: A review on absolute visual localization for uav. Robotics and Autonomous Systems 135, 103666 (2021). doi:10.1016/j.robot.2020.103666
9. Garcia Carrillo, L.R., et al.: Combining stereo vision and inertial navigation system for a quad-rotor uav. Journal of Intelligent and Robotic Systems 65, 373–387 (2012). doi:10.1007/s10846-011-9571-7
10. Yang, B., Yang, E.: A survey on radio frequency based precise localisation technology for uav in gps-denied environment. Journal of Intelligent and Robotic Systems 103 (2021). doi:10.1007/s10846-021-01500-4
11. Meles, M., et al.: Measurement based performance evaluation of drone self-localization using aoa of cellular signals. In: 2021 24th International Symposium on Wireless Personal Multimedia Communications (WPMC), , pp. 1–5. (2021)
12. Zhao, Y., et al.: Accurate 3d localisation of mobile target using single station with aoa–tdoa measurements. IET radar, sonar and navigation. 14(6) (2020-06)
13. Zhou, H., et al.: Imu dead-reckoning localization with rnn-iekf algorithm. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), , pp. 11382–11387. (2022)
14. ACEINNA, I.: Aceinna navigation studio: Extended kalman filter. https://openimu.readthedocs.io/en/latest/algorithms.html (2018)