Analysis of Sensor Fusion Solutions for UAV Platfoms

Jesus Garcia
Universidad Carlos III
Applied Artificial Intelligence Group
Madrid, Spain
jgherrer@inf.uc3m.es

Jose M. Molina
Universidad Carlos III
Applied Artificial Intelligence Group
Madrid, Spain
molina@ia.uc3m.es

***Abstract*—Evaluation of Unmanned Aerial Vehicle (UAV) systems is mostly based on simulation tools that are manually configured to analyse the system output. In this work, the authors present an original method to evaluate the perfor-mance of UAV platform in real situations based on available data. The main innovation is an evaluation for designing sensor fusion parameters using real performance indicators of accuracy of navigation in UAVs based on PixHawk flight controller and peripherals. This platform allows physical integration of the main types of sensors in UAV domain, and at the same time the use of powerful simulation models developed with Gazebo. This methodology and selected performance indicators allows to select the best parameters for the fusion system of a determined configuration of sensors and a predefined real mission.**

Keywords— UAVs sensor fusion, EKF, Real Data Analysis, System Design.

I. Introduction

Unmanned vehicles must be able to control their attitude and position by means of automatic control algorithms, they are controlled by a computer that integrates data from some electro-mechanical sensors and any local or global positioning system and applies any output control system to change its location using any locomotion system. The controller usually is an embedded microcontroller with appropriate interfaces to all vehicle components. As an example, PixHawk Px4 system integrates navigation data and software modules including fusion algorithms [1].

The main problem in navigation focuses on improving GPS with the ability to provide accurate navigation output when GPS data becomes unavailable due to unexpected outages or intentional problems (jamming or spoofing) in certain environments. Therefore, an approach based on the fusion of complementary sensors is essential, resorting to the fundamental equations of navigation and the characterization of the errors committed by each data source. This area has become popular due to the ubiquity of GPS and the availability of inertial sensors based on inexpensive MEMS components [2], [3], [4]. The integration of these complementary technologies allows compact and robust navigation solutions to determine attitude and location, so that the vehicle can determine its state in a robust way and use appropriate control techniques for autonomy. Other more drastic options for non-dependence on the GPS signal involve the deployment of autonomous localization systems such as the recognition of the environment by artificial vision [5] or location by means of electromagnetic beacons [6], with the associated cost of developing a complementary infrastructure.

Complementary to navigation technologies, the use of lasers in combination with other range detection sensors (sonar, radar, video), allows to extend the navigation conditions and obstacle avoidance**.** In order to develop obstacle avoidance algorithms, it is interesting to include software simulation to carry out tests without incurring risks for people and devices. Gazebo is a powerful 3D simulation environment for autonomous vehicles that is particularly suitable for testing object-avoidance. Gazebo can be used with Software In Loop (SIL) and Hardware In Loop (HIL) design. However, the simulation software will be outside the scope of the document. In the air vehicles (UAVs), the integration requirements (consume, weight, dimensions) are much more restrictive but, even so, it is a line in continuous development [7],[8],[9].

Therefore, research of robust and general techniques to integrate complementary data sources has become essential for this type of systems. In addition to theoretical developments, it is of vital importance the availability of equipment and experimental environments to validate the robustness of the solutions working in real-world conditions. The integration of sensors has to be based on the definition of parameters of the tracking system that should be adjusted to improve performance in a predefined set of missions with a defined set of sensors. The methodology proposed in this paper assumes that real system adjustments will be based on a real platform with predefined flight missions so that, in this context, the best parameters could be obtained analyzing the real operation of sensors and real output. Simulation of UAV environments is a powerful tool but not enough to evaluate in a thorough way these systems in real conditions of real missions. Accordingly to [13], evaluation tasks should be aligned with the user needs and how the fusion system meets the specifications. The selection of parameters and quality metrics is a complex task, particularly in real applications, since there are not ground truth or a standard methodology for making the data fusion evaluation. There are numerous examples of output analysis of algorithms and configurations based on simulation, such as characerizing navigation errors [14], sensor fault detection [15], or sensor integration in maritime navigation domain [16]. Other works in

UAV navigation use experimental real data sets, usually in a single flight, in order to assess specific aspects such as robustness against GPS outages [17] or impact of outliers in different solutios [18].

This paper presents the selected platform, design tools and environment for real experimentation, the proposed methodology and a selection of available data sources and effects of data processing techniques on the quality of the navigation solution. The main contribution of this paper is the briefing of a methodology used for adapting filter parameters to real conditions, and further systematic analysis of available real data. Section II introduces the selected working platform, detailing the architecture of its software and the vehicles we have made to test its capacities and collect data. Section III presents the proposed methodology and evaluation metrics, section IV explains experimental environment and and analysis of the PixHawk Px4 system filter and fusion algorithms following the presented methodology. Finally, section V summarizes the conclusions derived from this work.

II. The PixHawk and PX4 UAV system

A. Architecture

Unmanned vehicles must be able to control their attitude and position by means of automatic control algorithms. They are controlled by a computer that integrates data from some electro-mechanical sensors and any local or global positioning system, and applies any output control system to change its location using any locomotion system. This controller is usually an embedded microcontroller that performs the core of all vehicle components.

This research is based in the study of the PixHawk flight controller performance. An open-hardware computer designed by 3D Robotics specifically to create autopilot vehicles, that arises from the combination of PX4FMU and PX4IO boards. Both cards, from their version v2, are integrated in the same PCB (Printed Circuit Board) giving origin to PixHawk.

B. Sensors and data sources

The PixHawk board has several sensors integrated, shown in Table I, which serve as data sources to the PX4 stack and include some processing functions.

TABLE I. SENSORS INTEGRATED IN THE PIXHAWK BOARD

Sensor	*Type*	*Axes*	*Scale*	*ADC accuracy*	*Data rate*
L3GD20H	gyroscope	3	2000 dps	16 bits	760 Hz
LSM303D	accelerometer/ magnetometer	6	± 16g / ± 2gauss	16 bits	1600 Hz/ 100 Hz
MPU-6000	accelerometer/ gyroscope	6	± 16g / 2000 dps	16 bits	1000 Hz/ 8000 Hz
MS5611	barometer	1	1200 mbar	24 bits	1000 Hz

These sensors allow enhancing navigation capabilities and increase the accuracy of the stabilization system measurements, what is quite important when we want to create an unmanned vehicle, because allows a more faithful image of the flying environment.

C. Software for Flight Control and Data Processing

PX4 is the control software of PixHawk processor. It is a real-time operating system based on NuttX[1] and consists of two main layers: PX4 Flight Stack and PX4 Middleware. PX4 Flight Stack is the complete collection of applications embedded in PixHawk hardware for drone control, while PX4 Middleware is the interface that allows the flow of data from sensors to applications through a publish/subscribe system called uORB. uORB allows to publish the data coming from the sensors and make them available to the applications of the Flight Stack, obtaining a reactive system and totally parallelized. The outstanding modules are flight controller and sensor data processing [19].

Regarding the data processing, Px4 implements an AHRS (attitude and heading reference system) that implements different algorithms to estimate the vehicle attitude and creates a direction vector that allows the unmanned displacement. In this section we will overview some basic algorithms that run into the system during the flights.

a) Direction Cosine Matrix (DCM)

This program allows the analysis of the triaxial accelerometers and gyroscopes data to obtain a Direction Cosine Matrix [20]. It makes possible the conversion of real-time measurements into instantaneous orientation parameters of the vehicle to deliver roll, pitch and yaw angles or variations:

b) Inertial Navigation System (INS)

This algorithm calculates the trajectories and corrections that allows the vehicle to move between single points using the DCM data. It is used to estimate the vehicle attitude with high frequency, so it is especially useful to complement the global position obtained from the GPS data.

c) Extended Kalman Filter (EKF)

All measurements are affected by noise that should be taken into account in the estimation of attitude and cinematic parameters. The Px4 system counts with several Extended Kalman Filter algorithms to process all sensor data in a compensation function that depends of the specific noise and accuracy characterization of each sensor, throwing high accuracy estimations of the vehicle attitude. The Px4 application counts with the possibility of applying different EKF solutions running in parallel, using different sensor measurements and states. With this implementation, it is possible to increase the accuracy and consistence of estimates even if the vehicle losses the GPS signal in certain time intervals. Table II shows the three different available EKF modes.

D. Possibility of SIL and HIL design.

[1] www.nuttx.org

Pixhawk supports SIL [29] and HIL [30] using Gazebo simulation This way, it is possible to debug navigation and object-avoidance algorithims in PixHawk fligh controller, without using any real device. In this way, you could save on material cost, as well as increase the quality of the final product.

TABLE II. PX4 EXTENDED KALMAN FILTERS

Name	*Specification*
EKF1	Only use the DCM for attitude control and the Inertial navigation for AHRS reckoning for position control
EKF2	Use the GPS for 3D velocity and position. The GPS altitude could be used if barometer data is very noisy.
EKF3	If there is no GPS, it can use optical flow to estimate 3D velocity and position.

In addition, Gazebo offers various models of real autonomous vehicles, saving modeling time. In case of not being offered by Gazebo, it will be necessary to use time and resources in a correct physical modeling, so that the conditions of simulation could be as close as possible to reality. It will be the task of the engineer to evaluate the interest in using software simulation. Furthermore, it is necessary to keep in mind the fact that Sotfware modeling and simulation is only of interest when it is carried out by an experienced engineer in the field. Otherwise, the behavior of the system would not be reproduced in a faithful manner.

Specifically, in the case of Pixhawk, a HIL configuration allows running the code written in the flight controller without using any real sensor. That fact allows a first contact in the study of the effect of the parameters of navigation and evasion algorithms.

III. Evaluation of UAV sensor fusion in real conditions

In many real problems, simulated environments are used to define UAV sensors and the system parameters to optimize the system performance [14],[15],[16],[21],[22]. Some problems appear with this kind of methodologies, basically how to represent in simulation all effects appearing in real conditions and the way to evaluate the parameters configuration. UAV simulation have been applied to design the control subsystem for predefined missions, but simulation of real sensors is a major problem in this kind of approach. Real UAV conditions are not easy to model in simulators. UAV are affected for atmospheric conditions and random movements of UAV platforms, so accurate simulation of input data is extremely complicated for designing system parameters. As mentioned before, it would be necessary to evaluate advantages and disvantages of using simulation enviroments.

In this scenario, the proposed methodology, depicted in Fig. 1, tries to test the system parameters under real conditions. The first step of this methodology is the definition of the UAV platform, the type, cinematic characteristics, set of sensors and the tracking algorithms. Once the UAV platform is defined, the methodology is composed by the following steps:

1.- Mission definition. Parameters are selected from a set of possible values for a specific mission, defined by means of several waypoints. These waypoints are used to repeat the same mission every time that a new evaluation is done.

2.- Each time a flight of the UAV passing through the predefined waypoints (mission) is carried out, the values of the sensor data are stored (position and velocity taken from GPS, inertial data, magnetometers, etc.)

3.- A set of flights, with the same waypoints defining the mission, are carried out and the corresponding sets of sensors data are stored together with system output, using several configurations of parameters for filtering. These values are postprocessed offline.

4.- The best configuration of parameters is selected, analyzing the performance metrics for the set of missions carried out over the same waypoints.

5.- The selected parameters are introduced in UAV system to perform real mission

The decision about "the best" parameters should be based in a set of indicators to evaluate the quality of the main components of the data fusion system. The validation and quality assessment of fusion system is a fundamental step in the development. However, as indicated in [23], the development of objective evaluation metrics with no available ground truth is a challenge yet for data fusion researchers. There are no well-established procedures to systematically evaluate sensor fusion systems beyond simulated conditions, making in many times difficult to predict performance in real-world conditions. After a revision of previous works, there are scarce global metrics without ground truth of fusion system, such as [24] where the metrics are fusion break rate, rate of fusion tracks and track recombination rate. This terminology considers "global" metrics as those assessing the global fusion system output, while "local" metrics evaluate specific outputs from individual sensor data processes in a decentralized fusion architecture. Some local metrics without ground truth are: rate of non-associated data, rate of premature deleted tracks and average residual [24], association performance metrics for track purity and track switches [25], or number of missed targets, track life time, rate of false alarms, rate of track fragmentation and track latency [26].

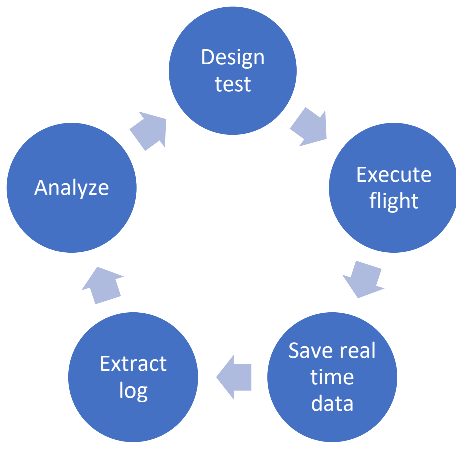

Fig 1. Evaluation methodology

TABLE III. TEST PERFORMED OVER THE SAME CIRCUIT

Name	*Specification*
Static test without propellers	Accelerometers and gryroscopes, noise.
Static test on ifle	Accelerometers and gryroscopes, noise
Unmanned flight test	GPS, inertial navitation system
Unmanned hold test	GPS, Flow Sensor, DCM
Manual flight test	PID and configuration parameters

In navigation function, since data association is not a problem (all sensor data associate to the vehicle track), the sensor fusion performance is evaluated with the following global indicators:

- averaged innovations
- fusion break rate

These metrics allow the validation of the designed system and decision of appropriate configuration parameters in complex scenarios.

1) Averaged innovations

The innovation, or residual, is computed in the tracking filter each time an update is done for the prediction. For instance, for horizontal XY position, denoting with sub-index p the predicted track and sub-index m for measurement, and considering the average along a time window with N_T measurements, it is defined as:

$$r = \frac{1}{N_T}\sum_{i,k}\left(\hat{x}_{pi}[k] - x_{mi}[k]\right)^2 + \left(\hat{y}_{pi}[k] - y_{mi}[k]\right)^2 \tag{1}$$

The normalized innovation also employs the covariance matrices considering both the predicted and observation uncertainties, matrix S. The averaged value of normalized innovations defined as:

$$r_n = \frac{1}{N_T}\sum_{i}\left[\hat{x}_{pi}[k] - x_{mi}[k] \quad \hat{y}_{pi}[k] - y_{mi}[k]\right] S_i[k]^{-1}\begin{bmatrix}\hat{x}_{pi}[k] - x_{mi}[k]\\ \hat{y}_{pi}[k] - y_{mi}[k]\end{bmatrix} \tag{2}$$

This value is a-dimensional, and represents the discrepancy between observations and predictions, averaged along the measurements contained in N_T.

Sometimes, if full covariance matrices are not available, a simplification is done and only the variances (diagonal terms) are considered:

$$r_n = \frac{1}{N_T}\sum_{i}\left(\frac{\left(\hat{x}_{pi}[k] - x_{mi}[k]\right)^2}{\sigma_{xp}^2 + \sigma_{xm}^2} + \frac{\left(\hat{y}_{pi}[k] - y_{mi}[k]\right)^2}{\sigma_{yp}^2 + \sigma_{ym}^2}\right) \tag{3}$$

2) Fusion break rate

The rate of fusion break, t_{FB}, is the number of times some navigation source is declared as inconsistent in the integrity analysis and therefore de-fused (the less consistent component is removed from the system track). This may lead to tracker re-initialization or keep the system track with a component less, the faulty sensor. It is computed as:

$$t_{FB} = \frac{\sum_i\{gt_i \mid gt_i \text{ is inconsistent track}\}}{N_T} \tag{4}$$

The value, averaged along N_T measurements, is obtained counting the total number of fusion break events during the interval. The test to decide the removal of a data source is done using the innovation, this time normalized by a higher number of standard deviations, typically 5.

IV. Experimental environment and evaluation of tracking filters for sensor fusion

The process of data acquisition was based on several flight test missions that have taken place on circuits like the one sketched in Fig. 2. For each specified mission, we applied different configuration parameters to analyze the performance differences between each setting up of tracking filters. The flight controller log data was saved together with the configuration settings to be analyzed. For instance, table III, shows different tests carried out on the previous circuit and the analyzed elements of the navigation system (sensors and controllers). Remember, for information purposes that HIL strategy could be used in order to get simulated sensor data instead of real data.

The most typical task carried out by the data fusion process of the Pixhawk (EKF2 filter) is the attitude estimation using magnetometer, gyroscope and accelerometer data (attitude and heading reference system), and then fuse with accelerometers and GPS data to estimate position and velocity.

So, the sensor fusion system is based on a loosely coupled architecture which uses GPS position and velocity measurements to aid the INS, typically used in most of navigation solutions based on sensor fusion [15],[18],[19],[21],[28]. In this way, the IMU sensors are used extrapolate position, velocity, and attitude at high frequency (50 Hz), while updates from GPS measurements at low frequency (1 Hz) allows refinement of cinematic estimates and inertial sensor biases. Typically, the estimated state vector resulting in the output for the GNSS/INS filter contains the attitude vector represented with a quaternion, 3D position and velocity, three gyro biases and three accelerometer biases. The selected coordinate frame for position and velocity is the ENU frame (East, North, Down) with respect to the tangent plan with origin defined by the arming point in the start of the mission.

The EKF filter, used for sensor fusion in navigation, depends on two sets of parameters wich are sensor noise and plant noise. Both the estimation of cinematic parameters and sensor biases critically depend on the parameters characterizing noise in sensor data and uncertainty in prediction (process noise). Speficicallly, process noise parameters affect to the predicted error covariance and have critical impact in the weights given to the sensor observations with respect to the predicted estimates. A higher value for these parameters imply higher values of predicted covariance and so higher gain to observations (since the confidence on prediction decreases). Conversely, lower values imply lower gain to observations (higher confidence on predictions). The first set (sensor noise parameters) is usually given by accuracy tables from sensor providers.

TABLE IV. PROCESS NOISE PARAMETERS OF EKF

Name	*Specification*	*Flight 1*	*Flight2*	*Flight3*	*Flight4*	*Flight5*	*Flight6*
EKF2_ACC_B_NOISE	Process noise for IMU accel. bias prediction	0.003 m/s^3	0.001 m/s^3	0.003 m/s^3	0.007 m/s^3	0.01 m/s^3	0.003 m/s^3
EKF2_GYR_B_NOISE	Process noise for IMU rate gyro bias prediction	0.001 rad/s^2	0.001 rad/s^2	0.003 rad/s^2	0.007 rad/s^2	0.01 rad/s^2	0.001 rad/s^2
EKF2_ACC_NOISE	Accelerometer noise for covariance prediction	0.35 m/s/s	0.1 m/s/s	0.3 m/s/s	0.7 m/s/s	1.0 m/s/s	0.35 m/s/s
EKF2_GYR_NOISE	Rate gyro noise for covariance prediction	0.015 rad/s	0.01 rad/s	0.03 rad/s	0.07 rad/s	0.1 rad/s	0.015 rad/s
EKF2_ACC_B_NOISE	Process noise for IMU accel. bias prediction	0.003 m/s^3	0.001 m/s^3	0.003 m/s^3	0.007 m/s^3	0.01 m/s^3	0.003 m/s^3

A. GPS and INS local position integration

Fig 2. Mission circuit design

With respect to process noise, the values which can be tunned in the available platform have been systematically analyzed in the six scenarios used, with flights repeating the programmed mission (waypoints), but changing the parameters affecting to EKF performance shown in table IV. The selected values, to analyze the impact on performance metrics appear also in table IV and have been set considered the minimum and maximum values recommended in the implemented EKF2 system.

Fig. 3 presents a zoom of horizontal position estimated by tracking filter and GPS observations (circles), corresponding to right-bottom corner of mission. As can be appreciated, the flights corresponding to higher values of parameters (like flight 1, in blue, flight 5, in black) present lower deviations during turns, and, conversely, are affected more by the sensor noise.

B. Analysis of innovations

The normalized innovations are presented in this section. Fig. 4 presents aggregated position and velocity in a 6D innovation vector, normalized by its covariance matrix. In this case can be seen that flights 4,5 present the lower values. Table VI summarizes the innovation analysis for the 6 flights. The results showed in detail the system output with different parameters, reflecting the impact assessed through different magnitudes and the quality metrics considered (averaged innovations and fusion breaks).

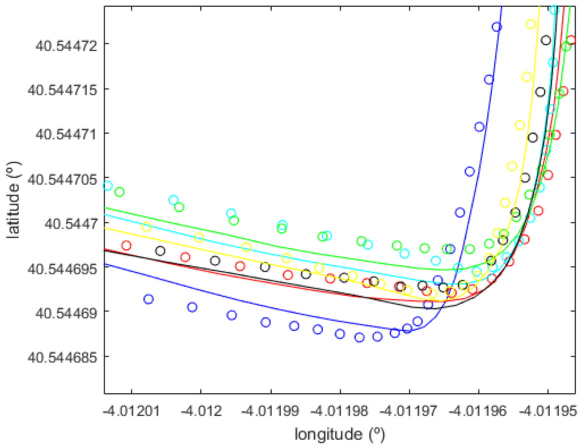

Fig. 3: Details of lat-lon GPS input and EKF output for all flights (1-blue, 2-red, 3-cyan, 4-green, 5-black, 6-yellow)

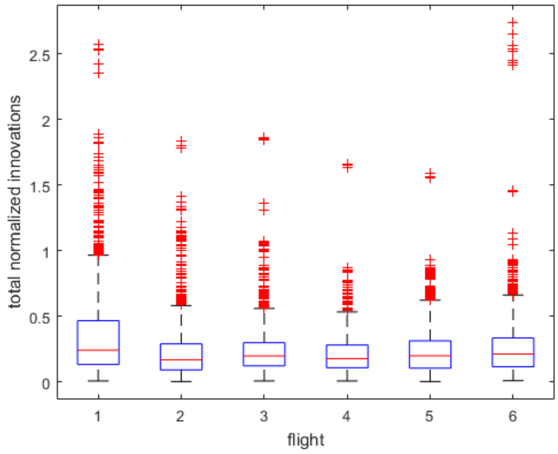

Fig 4. Normalized innovations for all flights

The methodology allows to take decisions of appropriate parameters for the mission considering the global residual of navigation vector and difference with respect to the default configuration, corresponding to the default configuration (flight #4). A moderate improvement in averaged residuals was appreciated, while the robustness in terms of fusion breaks, is not critically dependent on these parameteres within the recommended intervals.

Finally, the methodology can be generalized and applied to different missions and sets of available sensors to explore the sensitivity of fusion algorithms and find the optimal parameters.

V. Conclusions

This paper presented a platform (Pixhawk PX4) and methodology to experiment with real data for UAV navigation. Based on data analysis and characterization the algorithms can take advantage of available sources. The quality of all inputs was systematically analyzed, and three processing algorithms, DCM, LPF and EKF, were evaluated with different parameters to exploit the data in the appropriate way considering the output analysis and specific performance metrics not based on ground truth.

So, this work presents a methodology to test and configure UAV navigation systems in real conditions, illustrated with an open environment for experimentation. The analysis of real data in a systematic way will allow successive improvements and parametrization, considering, among others, the following aspects:

- Data filtering to reduce errors and remove outliers
- Quality analysis to weight data uncertainty
- Analysis of biases and calibration previous to fusion.
- Parameter adjustment to optimize performance (PID gains, filter parameters, observation and plant noises, etc.)

Acknowlegments

This work was funded by public research projects of Spanish Ministry of Economy and Competitivity (MINECO), references TEC2017-88048-C2-2-R, RTC-2016-5595-2, RTC-2016-5191-8 and RTC-2016-5059-8.

REFERENCES

[1] Garcia J. Molina J.M., J. Trincado. "Analysis of real data with sensors and estimation outputs in configurable UAV platforms". Sensor Data Fusion: trends, solutions and applications 2017. Bonn, Germany, 10-12 Oct. 2017.
[2] Groves P.D. "Navigation using inertial sensors". IEEE AES Magazine (Vol.30, Iss. 2) 42 - 69 Feb. 2015
[3] Britting, K.R. "Inertial navigation systems analysis". Artech House, 2010
[4] Farrel J.A., "Aided Navigation: GPS with High Rate Sensors", McGraw-Hill, New York, 2008.
[5] Choi H., Kim Y.. "UAV guidance using a monocular-vision sensor for aerial target tracking". Control Engineering Practice, 22, 10-19, 2014.
[6] Torres-González A., Martínez-de Dios J.R., Ollero A. "Robot-Beacon "Distributed Range-Only SLAM for Resource-Constrained Operation". Sensors 2017, 17(4), 903.
[7] Ferrick A., Fish J., Venator E. and Lee G.S. "UAV Obstacle Avoidance Using Image Processing Techniques Technologies for Practical Robot Applications (TePRA", 2012 IEEE International Conference on 23-24 April 2012
[8] Fasano G., Accado D., Moccia A., Moroney D., "Sense and avoid for unmanned aircraft systems". IEEE Aerospace and Electronic Systems Magazine, vol. 31, nº 11, pp. 82-110, 2016.
[9] Rankin G., Tirkel A., Leukhin A. "Millimeter Wave Array for UAV Imaging". MIMO Radar Radar Symposium (IRS), 2015 16th International 24-26 June 2015
[10] Broumandan A., Siddakatte R, Lachapelle G. "An Approach to Detect GNSS Spoofing". IEEE AESS Magazine Vol. 32, N8, August 2017. 64-75

TABLE V. RMS OF INNOVATIONS

Variable	***Flight 1***	***Flight 2***	***Flight 3***	***Fligt 4***	***Flight 5***	***Flight 6***
Pos_x	0.1456	0.1786	0.2011	0.1958	**0.1205**	0.1684
Pos_y	0.1366	0.1220	0.1451	0.1368	0.1294	**0.1070**
Pos_z	0.1674	0.1468	0.1452	**0.1414**	0.1527	0.1651
Vel_x	0.3495	0.2583	0.2280	**0.2059**	0.2517	0.2746
Vel_y	0.3250	0.2276	0.2187	**0.2025**	0.2732	0.2610
Vel_z	0.2115	**0.1940**	0.2226	0.2192	0.2060	0.2172
Aggregate d pos-vel	0.4574	0.3040	0.2878	**0.2532**	0.2926	0.3456

[11] Xiao L., "GNSS Receiver Anti-spoofing Techniques: A Review and Future Prospects". 5th Electronics and Network conference (CECNet 2015) Shanghai, 2015.
[12] Carson N., Martin S.M., Starling J., and Bevly D.M. "GPS Spoofing Detection and Mitigation Using Cooperative Adaptive Cruise Control System". 2016 IEEE Intelligent Vehicles Symposium (IV) Gothenburg, Sweden, June 19-22, 2016. 1091-1096
[13] Blasch E., Pribilski M., Daughtery B., Roscoe B., and Gunsett J., "Fusion Metrics for Dynamic Situation Analysis," Proc. SPIE 5429, April 2004.
[14] Kim J., Cheng J., Guivant J., Nieto J., "A. Compressed Fusion of GNSS and Inertial Navigation with Simultaneous Localization and Mapping". IEEE AESS Magazine Vol. 32, N8, August 2017. 22-36
[15] Sun R., Cheng Q., Wang G. and Ochieng W.Y. "A Novel Online Data-Driven Algorithm for Detecting UAV Navigation Sensor Faults". Sensors 2017, 17(10), 2243;
[16] Gao W., Zhang Y. and Wang J." A Strapdown Interial Navigation System/Beidou/Doppler Velocity Log Integrated Navigation Algorithm Based on a Cubature Kalman Filter". Sensors 2014, 14(1), 1511-1527
[17] Yao Y. and Xu X. "A RLS-SVM Aided Fusion Methodology for INS during GPS Outages". Sensors 2017, 17(3).
[18] Jiang C., Zhang S.B. and Zhang Q.Z. "A New Adaptive H-Infinity Filtering Algorithm for the GPS/INS Integrated Navigation". Sensors 2016, 16(12), 2127
[19] Babister, A. W. (1980). "Aircraft dynamic stability and response" (1st ed.). Oxford: Pergamon Press. ISBN 978-0080247687.
[20] Hyyti H. and Visala A., "A DCM Based Attitude Estimation Algorithm for Low-Cost MEMS IMUs". International Journal of Navigation and Observation, Volume 2015, Article ID 503814.
[21] Martí E., García J., Molina J.M. "A Simulation Framework for UAV Sensor Fusion". Hybrid Artificial Intelligence Systems 5th International Conference, HAIS 2010, San Sebastián, Spain, June 23-25, 2010.
[22] Layh T., Demoz G.E."Design for Graceful Degradation and Recovery from GNSS Interruptions", IEEE AESS Magazine Vol. 32, N9, September 2017. 4-17.
[23] Khaleghi B., Khamis A., Karray F.O., Razavi S.N., "Multisensor data fusion: A review of the state-of-the-art". Information Fusion Volume 14, Issue 1, January 2013, Pages 28–44
[24] García, J., Luis, Á., Molina, J. M. "Quality-of-service metrics for evaluating sensor fusion systems without ground truth". 19th International Conference on Information Fusion, Heidelberg, Germany, July 2016.
[25] Colegrove, S. B., Bldg, I. B., Cheung, B., & Davey, S. J. (2003). "Tracking System Performance Assessment". 6th International Conference on Information Fusion, 926–933, Queensland, Australia, July 2003
[26] Gorji, A.A., Tharmarasa, R., and Kirubarajan, T. "Performance measures for multiple target tracking problems". 14th International Conference on Information Fusion. Chigaco, USA, July 2011.
[27] Meier L., PX4 Development Guide. [Online]. Available: https://dev.px4.io/en/
[28] Martí E., García J., Escalera A., Molina J.M., Armingol J.M. "Context-Aided Sensor Fusion for Enhanced Urban Navigation". Sensors Vol. 2012 16802-16837. 2012
[29] Dronecode. PX4 Development Guide [Online]. Available: https://dev.px4.io/en/simulation/gazebo.html