Autonomous Track and Follow UAV for Aerodynamic Analysis of Vehicles

Ahmad Drak and Alexander Asteroth
Department of Computer Science, Bonn-Rhein-Sieg University, Germany

Abstract: *This work addresses the issue of finding an optimal flight zone for a side-by-side tracking and following Unmanned Aerial Vehicle(UAV) adhering to space-restricting factors brought upon by a dynamic Vector Field Extraction (VFE) algorithm. The VFE algorithm demands a relatively perpendicular field of view of the UAV to the tracked vehicle, thereby enforcing the space-restricting factors which are distance, angle and altitude. The objective of the UAV is to perform side-by-side tracking and following of a lightweight ground vehicle while acquiring high quality video of tufts attached to the side of the tracked vehicle. The recorded video is supplied to the VFE algorithm that produces the positions and deformations of the tufts over time as they interact with the surrounding air, resulting in an airflow model of the tracked vehicle. The present limitations of wind tunnel tests and computational fluid dynamics simulation suggest the use of a UAV for real world evaluation of the aerodynamic properties of the vehicle's exterior. The novelty of the proposed approach is alluded to defining the specific flight zone restricting factors while adhering to the VFE algorithm, where as a result we were capable of formalizing a locally-static and a globally-dynamic geofence attached to the tracked vehicle and enclosing the UAV.*

Keywords: *UAV, flight zone, geofence, dynamic vector fields, aerodynamics.*

Received October 6 2018; accepted January 22 2019

1. Introduction

Efficient design and evaluation of aerodynamically optimized shapes and/or shells of vehicles is an overwhelming and demanding task that requires several iterations of simulating and testing under different conditions. Certain aerodynamic properties of bodies cannot be easily acquired in a wind tunnel test or through simulation. In order to be able to extract this information from the real airflow we propose to first attach tufts made of wool to the sides of the vehicle. Image analysis techniques can then be used to extract the positions, orientation, and deformations of the tufts over time. This tuft data will then be used to reconstruct the airflow around the vehicle, enabling comparative tests of various vehicle exteriors under real-world conditions.

One challenge associated with such a method is acquiring images of the tufts without influencing the airflow of the vehicle. Due to this problem, it is insufficient to mount the camera on the vehicle itself, as this causes turbulence at the surface and the airflow to detach. Such turbulence will cause misinterpretation of the extracted tuft data, and consequently the disturb the airflow of the vehicle. The vehicle used as a testbed for this work can be seen in Figure 1.

The proposed method to solve this problem is by using an Unmanned Aerial Vehicle. The Unmanned Aerial Vehicle (UAV) is equipped with necessary sensors and a gimbled camera, which allows it to track and follow the vehicle autonomously, while obtaining high quality video of the tufts. There should be no interaction between the generated thrust from the drone and the tufts attached to the vehicle, as this will influence the airflow around it. The video recorded can be used to extract relevant information from the tufts by means of the VFE algorithm. The use of a UAV in this scenario is paramount as it allows to traverse the real world with little restrictions, as compared to a ground vehicle. In this sense, the UAV would not be restricted to traversing on a road, rather it has all the space needed to accurately follow and track the vehicle. Moreover, other alternatives to the use of a UAV, such as a car, bike or otherwise, are not feasible due to space limitation caused by side-by-side tracking, in addition to the uneven road which causes additional unwanted vibrations.

This paper is aimed towards finding an optimal flight zone for a UAV to perform side-by-side tracking and following of an observed vehicle, as adhering to several space-restricting factors inherited from the extraction of dynamic vector fields from tufts attached to a vehicle. The novelty of the proposed method is alluded to dynamically constructing a local geofence enclosing the UAV at real-time, as adhering to the three major space-restricting factors. Moreover, the geofence is designed to be static with respect to (w.r.t) the tracked object but dynamic w.r.t to the world, the coordinates of which are updated at every epoch as the vehicle is being actively tracked.

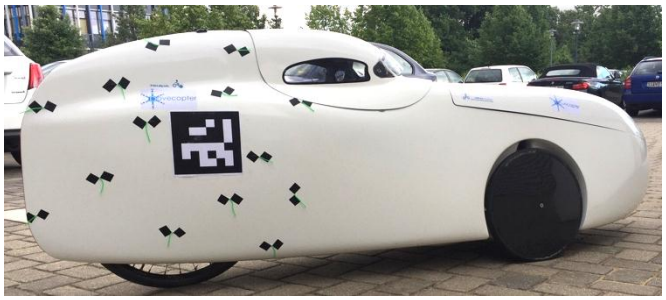

Figure 1.The lightweight testbed vehicle, seen with several tufts attached.

2. Related Work

Typical tests regarding the investigation of the aerodynamic properties of any vehicle is done by means of a wind tunnel or Computational Fluid Dynamics (CFD) simulation. A general problem with such methods is that they do not reflect a fully realistic environment for testing, since the effects of side winds, road architecture and external disturbances are generally ignored. Even though wind tunnel testing has been widely used in the past with a significant number of methods for aerodynamic analysis [8] its cost increases significantly with respect to the quality of results.

Moreover, these tests are idealized in the sense that they have a dominant airflow direction and little external turbulence. It is also observed that wind tunnels tend to have a ”regular” ground architecture, as compared to the real world’s “irregular” ground architecture, which is not a reflection of the real word conditions. This is of particular importance as certain aerodynamic properties are located on or near the area of the vehicle floor and wheels [10]. It is preferred, however, to use a wind tunnel to investigate the aerodynamics of an airplane. This is because airplanes fly at near Mach 1 speeds, which causes a change in the density of the surrounding medium, as a result the side winds can be considered negligible, which is not the case for very efficient and lightweight ground-based vehicles. In addition, as the outer-shell is made from lightweight materials, it can bend and buckle in the bind, altering its aerodynamics in ways that are difficult to model in simulation or idealized wind tunnel conditions.

CFD simulations provide more realistic environment conditions, but come at a high computational cost and a reduced accuracy. This is mainly due to the underlying approximation nature of CFD and a relatively low data resolution. There are several ways to reduce the computational cost of the CFD, such as using a meta-model [5].

Several techniques exist in literature regarding the tracking and following task of the UAV. Two general methodologies are identified, which are deterministic tracking and stochastic tracking [11]. Deterministic tracking approaches tend to reduce the problem to an optimization one where a cost function is chosen and minimized. A typical choice of cost function is the Sum-Of-Squared Distances (SSD) where a gradient descent algorithm is used to find the minimum [11].

This is seen in [2, 4] where the authors defined an SSD cost function between the observation and a fixed template, where the task is to find the parameter which minimizes the cost function. An alternative deterministic approach would be the mean shift, where the cost function is defined in terms of the colour histogram [2].

Stochastic tracking approaches are often reduced to an estimation problem where they tend to estimate the state(s) of a state-space model [11]. In the early work presented in [6], a Kalman filter was used for estimation, but it was soon realized that the type of model used was restricted. A modification to this was introduced by [1] where sequential Monte Carlo algorithms were used instead.

Siam and Elhelw [7] claim a novel framework which executes its tasks at real-time. The framework operates by utilizing image feature extraction and projective geometry. The outlier image features are computed and the moving targets are detected by using a spatial clustering algorithm. The targets are then tracked by using a Kalman filter with persistency check.

3. Flight Zone Factors

Gurriet and Ciarletta [3] defined their flight zone based on constrained geofence optimization in order to find a suitable controller input which prevents collisions. An optimal flight zone in this work will be defined by the bounds of a local geofence attached to the vehicle. A geofence is a static virtual geographic perimeter surrounding an object. A geofence operates by storing a list of Global Positioning System (GPS) coordinates that define the boundaries of this virtual fence, in addition to a maximum allowable altitude.

The geofence relies on flight data supplied by the flight controller’s sensors, such as GPS and Inertial Measurement Unit (IMU) readings [3]. Based on the sensors readings and the flight dynamics, the drone’s position is continuously checked against the list, and in case of a fence breach, the pilot is notified. For this particular application, a geofence is an essential component to achieve the global objective of providing high quality videos of the tufts. The geofence aids in defining and restricting the flight boundaries for the UAV, which adds to the overall safety factor and allows for restricting the tracking and following problem to a defined subspace. As opposed to the traditional global and static geofence, this particular local geofence is static with respect to the vehicle but dynamic with respect to the real world.

The vehicle will be tracked and followed based on tufts attached to its side, which due to the nature of

operation of the UAV, several factors restrict and define the size of the flight zone. As illustrated in Figure 2, the three major restricting factors are:

1. The distance between the UAV and the vehicle.
2. The angle of recording the video.
3. The altitude of the UAV.

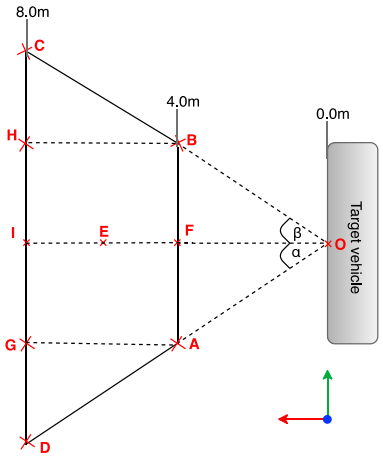

Figure 2. Top-view illustration of the geofence.

3.1. Distance

The distance factor can be further divided into two sub-factors:

- Minimum distance between the vehicle and the UAV with respect to thrust generated from the propellers. This is of particular importance as no interference from the generated thrust should interact with the tufts attached to the vehicle.
- Maximum distance between the UAV and the vehicle with respect to recording video of the tufts. A modified GoPro Hero4 camera with a 9mm lens is used to record the high quality videos of the tufts. In other words, this is the maximum distance away from the vehicle where the camera can remain capturing high quality videos which are accepted by the dynamic vector fields extraction algorithm.

3.2. Angle

The angle factor can also be further divided, however due to symmetry, the α and βangles are equal. The nominal placement of the drone and consequently the camera recording angle is perpendicular to the surface of the vehicle with the tufts attached. At the perpendicular, there is a zero degree angle between the camera Field Of View (FOV) and the tufts, hence it is in the best possible position. The zero degree angle is retained by taking the current position of the UAV as the centre of the geofence, indicated by point E in Figure 2. A deviation from this perpendicular could possibly lead the extraction algorithm to produce unreliable results, therefore, the maximum angle that the UAV can deviate from the perpendicular has to adhere to the limits of the extraction algorithm.

3.3. Altitude

Maintaining a valid range of altitude for the drone is mainly affected by the FOV of the camera. Within the virtual walls of the geofence, the FOV of the UAV should always include the tuft markers and the position marker used for tracking the vehicle. High altitudes lead to a potential loss of the desired FOV, while a low altitude will result in the unwanted ground effect, which in turn negatively affects the stability and performance of the drone.

4. Geofence Design

In order to design the geofence, and during the tests carried out to determine the values of the restricting factors, the cameras settings were chosen to be 2.7k resolution, 60 frames per second with a medium field of view[1]. The needed minimum distance between the UAV and the vehicle is found when there is no visible interaction between the generated thrust of the UAV and the tufts, which corresponds to the distance $\overline{OF}$ in Figure 2.

The maximum distance corresponds to the furthest distance away from the vehicle where the UAV is still capable of capturing HQ videos that adhere to the extraction algorithm, all the while maintaining the tracking and following of the vehicle. This corresponds to the distance $\overline{OI}$ in Figure 2.

The angle tests are performed to define the maximum value for the α and β angles seen in Figure 2. The vehicle surface is assumed to be flat in the location where the tufts are attached. As previously stated, the nominal angle for the drone is along the perpendicular to the vehicle, and hence due to symmetry, both α and $\beta$are equal. Therefore, it follows that the maximum deviation from the perpendicular $\overline{OF}$ where the drone can capture HQ video adhering to the extraction algorithm, is the maximum needed angle. The angle was acquired by hovering the drone at a fixed altitude and distance and varying the angle by 10° at each test. At each angle step, a video was recorded and checked against the extraction algorithm.

A total of 60 different tests were performed to determine the distance, angle and altitude, indoors and outdoors under different weather conditions. For each factor and for each test a video of the tufts was recorded and evaluated against the VFE algorithm.
The results of said tests are summarized in Table 1.

[1]These settings refer to the GoPro Hero 4 camera with a modified 9mm lens, as opposed to the standard wide-angle lens.

Table 1. Results of distance, angle and altitude tests performed to investigate the geofence design.

Factor	w.r.t	Minimum	Maximum
Distance	Propellers	4.0m	n/a
	Tuft recording	0.0m*	8.0m
Angle	Tuft recording	0.0°‡	45.0°
Altitude	Tuft recording	0.70/0.55/0.90<	0.87/1.8/1.25<

Note that * indicates any value lower than the minimum distance set by propellers, which is ignored. ‡ 0.0° implies that the camera POV is perpendicular to the vehicle's surface. While < indicates testing on $\overline{AB}$,$\overline{CD}$and point E, respectively.

With the performed tests and the values of the restricting factors defined, the flight zone can be defined as the ABCD zone in Figure 2 where the AOB zone is a no-fly zone.

To aid in constructing the geofence and to obtain the dimensions of all its sides, a Computer Aided Design (CAD) tool was used. The CAD tool allowed for a systematic way to draw the geofence, which in turn resulted in easily visualizing the flight zone, as seen in Figure 3.

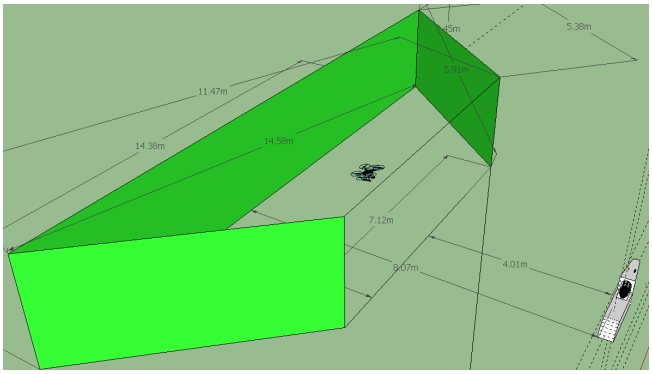

Figure 3. Visualization of the resulting flight zone after taking into account the restricting factors.

In order to construct the geofence programmatically and define its corner coordinates in terms of GPS units, the dimensions of all sides were obtained and summarized in Table 2. Therefore, by obtaining the current position of the drone via its GPS unit and assigning this coordinate as centre point E, it follows that the coordinates for the points A,B,C, and D can be found at each epoch due to the defined distances in Table 2. And hence, the geofence is virtually erected and serves as a local and static geofence w.r.t to the vehicle, but dynamic w.r.t the world.

Table 2. Geofence bounded flight zone parameters and their corresponding values. Refer to Figure 2 for the parameters.

Parameter	Value	Parameter	Value
α and β angles	45°	$\overline{DA}$	5.38m
$\overline{AB}$	7.12m	$\overline{OF}$	4.01m
$\overline{BC}$	5.38m	$\overline{OI}$	8.07m
$\overline{CD}$	14.38m	$\overline{OE}$	6.1m

5. Track and Follow Module

The geofence parameters tabulated in Table 2 are sufficient to construct the virtual fence and update its coordinates at real-time. The tracking and following task of the UAV is pertinent for its operation, as it determines the position of the target vehicle via a synthetic fiducial marker, predicts the next likely location of the vehicle through a linear estimator, and calculates the needed drone velocity to keep up with the tracked vehicle. In addition it performs safety checks that govern the integrity of the geofence structure, and the safety of the operator. A high-level overview of the operation of the track and follow module can be seen in Figure 4.

The real-time world coordinates of the drone are acquired via the Real-Time Kinematic (RTK) GPS unit onboard the drone. An ArUcomarkerattached to the tracked vehicle allows the drone to acquire the current world coordinates of the vehicle via an onboard HD camera. The joint rotation-translation matrix[R|t] containing the intrinsic and the extrinsic parameters of the camera, along with the radial and tangential distortion coefficients, transforms the coordinates of a 3D point to a coordinate system which is fixed w.r.t the camera, and hence, describes the motion of an object in front of the camera.

The 'getPositions' submodule in Figure 3 receives the current position of the UAV from the RTK GPS unit, along with the [R|t] matrix that is updated at every epoch. Consequently, the estimated position of the tracked vehicle is updated at every time step. This estimated position is then received by the next submodule to predict the next position of the vehicle.

After receiving the current coordinates of the UAV and the tracked vehicle, given sufficient knowledge of the system, the measuring sensor, uncertainty in the model, measurement errors and system noise, the Kalman filter can produce an optimal estimate of the system state, and predict the next likely position of the vehicle.

The filter utilizes a first-order constant-velocity model to predict the next state of the system at the following time step, in addition to adjusting the belief that accounts for the uncertainty in the prediction.

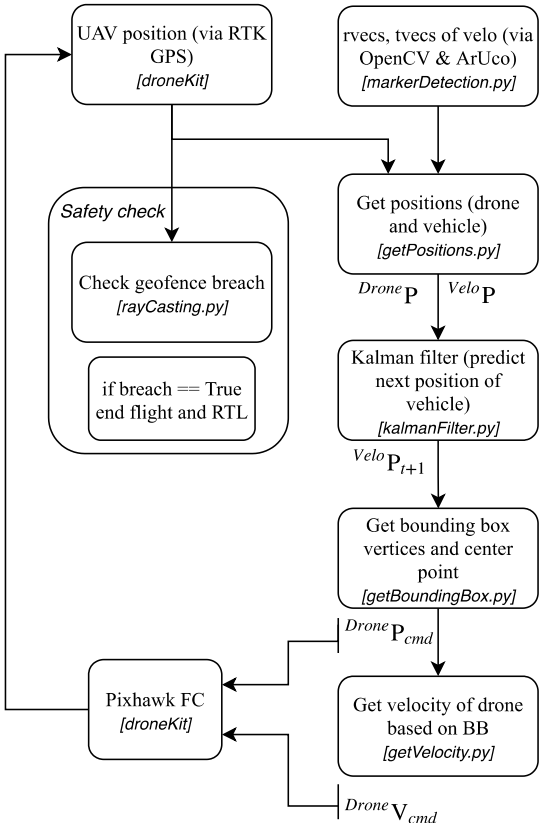

Figure 4. Track and follow module operation overview.

The next predicted position of the tracked vehicle is then received by the 'get bounding box' submodule, which in turn calculates the coordinates of the geofence vertices relative to the new position of the tracked vehicle at real-time. The parameters found and tabulated in Table 2 are also used in order to dynamically construct the geofence at every time step.

As soon as the next predicted position is obtained, the centre point coordinates (point E) of the geofence is calculated first, followed by the coordinates of points A,B, C and D (refer to Figure 2). The submodule calculates the change in latitude and longitude in both directions based on the centre point E and its offset distance from the vehicle ($\overline{OE}$), and returns the minimum and maximum of the latitude and longitude bounds, as seen in Equation (1). From the minimum and maximum bounds returned, the coordinates of the points can be formulated as seen in Equation (2). (The value of 110.57 represents the approximate distance for every degree longitude or latitude, and $(latitude_{cp})$ refers to the latitude coordinates of the centre point E).

$$\Delta\, latitude = \frac{offset\ distance}{110.57}$$
$$\Delta\, longitude = \frac{offset\ distance \cdot \cos(latitude_{CP})}{110.57} \quad (1)$$

$$\begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} = \begin{bmatrix} [longitude_{max} & latitude_{min}] \\ [longitude_{max} & latitude_{max}] \\ [longitude_{min} & latitude_{max}] \\ [longitude_{min} & latitude_{min}] \end{bmatrix} \quad (2)$$

Based on the acquired target position, current UAV position and the calculated coordinates of the geofence (Equation (2)), the submodule outputs the next commanded position of the UAV in order to remain within the bounds of the geofence while tracking and following the vehicle.

Moreover, the 'get velocity' submodule calculates a suitable commanded velocity for the UAV based on the absolute distance difference to the commanded position supplied by the 'get bounding box' submodule. Both the commanded velocity and the commanded position are sent to the drone where the flight controller reacts based on those commands.

In addition, several safety checks are in place in order to maintain the safety of the operator and/or any pedestrians. Furthermore, the safety check submodule implements a modified version of the ray casting algorithm to determine if the UAV breached the geofence. Upon a breach, the UAV is commanded to hover in place for five seconds and then Return To Launch (RTL).

The ray casting algorithm is a method used to solve the point-in-polygon problem, determining whether a point lies inside or outside the bounds of a defined polygon [9]. In this particular application, the coordinates of the UAV are periodically checked to be within the bounds of the geofence. However, rather than just casting a ray from the UAV to check for a breach, the modified version of the algorithm initially checks that the latitude and longitude coordinates of the UAV against the maximum and minimum longitude and latitude, since these values set the bounds of the geofence. If the UAV coordinates exceed the maximum and/or minimum limit, then it is classified as outside the geofence and no ray casting is needed, otherwise, ray casting is performed. This modification is added to speed up the breach checks since the safety submodule will be running continuously at twice the normal frequency of the system.

6. Discussions and Conclusions

This article presented work towards obtaining an optimal flight zone for a side-by-side tracking UAV. The optimality factor was the dynamic Vector Field Extraction (VFE) algorithm. The algorithm is supplied with HQ video recordings of tufts attached to a tracked vehicle, upon which it extracts the vector fields for each tuft at each frame.

Traditional testing methods, such as a wind tunnel or CFD simulations, do not reflect a true realistic environment due to the neglected side winds, road architecture and external disturbances. Moreover, the tracked vehicle is made from lightweight materials that tend to bend and buckle, which is hard to model in standard idealized tests.

The flight zone is restricted by three factors, distance, angle and altitude. For each of the factors countless tests were performed in order to determine

the minimum and maximum distance between the UAV and the vehicle, the maximum angle of recording the tufts and the maximum allowable altitude of the UAV. Upon defining these factors and finding their values, the static geofence could be created and its coordinates updated at every time step.

To construct the geofence, the longitude and latitude ([long,lat]) coordinates of the UAV are acquired via the RTK GPS unit on-board. These coordinates represent the centre point of the flight zone, point E in Figure 2. The coordinates of the corresponding points (A, B, C, and D) are then calculated in real-time based on a trapezoidal geometric equation, with a total execution time of 1.1ms for an average of 20 runs. Due to the relatively low execution time, the geofence is updated at the operating frequency of the flight controller at 250Hz. The algorithm is run on-board a dedicated companion PC on the UAV and connected physically to a dedicated flight controller.

The visualized geofence based on the calculated coordinates can be seen in Figure 5. The difference between the true values and the measured ones can be seen in Table 3. The error values are generally low with a maximum error of 0.82m in the distance between the centre point of the geofence and the vehicle ($\overline{EO}$). Even though it is comparatively higher than the other errors, its effect on the overall geofence structure is minimal since an offset of 0.82m along the axis perpendicular to the vehicle, still allows the UAV to remain within the bounds of the geofence. Overall, with distance errors ranging between 0.27-0.82 m, the integrity of the structure of the geofence still holds and its purpose of keeping the UAV within its bounds is not affected.

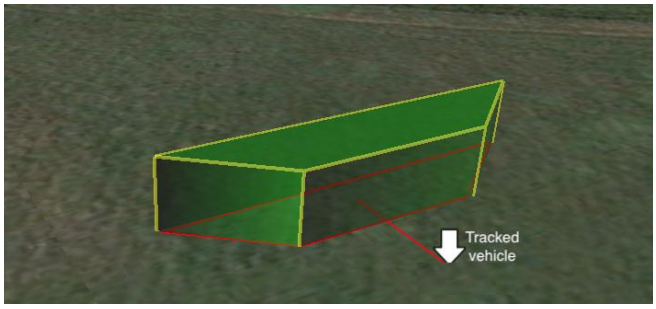

Figure 5. Visualized geofence enclosing the flight zone based on the [long,lat] coordinates of the UAV at the centre-point of the flight zone. (point E in Figure 2).

Moreover, the functionality of the marker detection was evaluated by calculating the reprojection error, which is the RMS error between the points in the real image and the points projected, based on the cameras intrinsic and extrinsic properties. The reported error of 0.89 pixels was deemed acceptable. The distance and angle factors are of grave importance for marker detection and the maximum for both was 8.0m and 45◦. The variance of the marker detection algorithm was tested by placing a marker and the camera in fixed locations and observe the output for 50 detections. The maximum reported variance was 0.0232.

Table 3. Ground truth and measured values of the various sides of the flight zone, in addition to the percentage error.

Distance	Ground truth(meters)	Measured value(meters)	Percentage error(%)
$\overline{AB}$	7.12	7.66	7.58
$\overline{BC}$	5.38	5.11	5.01
$\overline{CD}$	14.38	15.55	8.14
$\overline{DA}$	5.38	5.06	5.95
$\overline{OE}$	6.1	5.28	13.44

The possible sources of error are the position output of the marker detection algorithm used in tracking the vehicle, in addition to the geographic trapezoidal geometric equations that calculate the coordinates of points A, B, C, and D based on point E (coordinates of UAV).

The error caused by the marker detection is propagated to the geographic calculations since the reference point of the calculations is based on that acquired distance. In addition, the occlusion of the marker during a flight sends a null value to the modules down the pipeline, causing the submodules to crash intermittently. Moreover, errors could be caused by the spherical Earth model used in the calculations and the assumption that each degree corresponds to approximately 110 km, in addition to assuming an average Earth radius of 6371 km.

However, even with a maximum error of 13.5% the geofence structure is not affected and performs as expected with regards to keeping the UAV within its bounds. In the case of a fence breach, the UAV is autonomously commanded to hover at its current altitude and position for 10 seconds then safely land.

Acknowledgment

We are very grateful for financial support given by the Ministry for Innovation, Science and Research (MIWF) of the North Rhine-Westphalia state within the program FH-STRUKTUR 2017 (AZ: 322-8.03.04.02-FH-STRUKTUR 2017/07).

References

[1] Arnaud D., De Freitas N., and Gordon N., *Sequential Monte Carlo Methods in Practice*, Springer, 2001.

[2] Comaniciu D. and Ramesh V., "Mean Shift and Optimal Prediction for Efficient Object Tracking," *in Proceedings of International Conference on Image Processing*, Vancouver, pp. 70-73, 2000.

[3] Gurriet T. and Ciarletta L., "Towards a Generic and Modular Geofencing Strategy for Civilian UAVs," *in Proceedings of International Conference on Unmanned Aircraft Systems*, Arlington, pp. 540-549, 2016.

[4] Jang S., Choi K., Toh K., Teoh A., and Kim J., "Object Tracking based on an Online Learning Network with Total Error Rate Minimization," *Pattern Recognition*, vol. 48, no. 1, pp. 126-139, 2015.

[5] Kajero O., Thorpe R., Yao Y., Wong D., and Chen T., "Meta-Model based Calibration and Sensitivity Studies of CFD Simulation of Jet Pumps," *Chemical Engineering and Technology*, vol. 40, no. 9, pp. 1674-1684, 2017.

[6] Lee K., Yu Y., Wong K., and Chang Y., "Tracking 3-D Motion from Straight Lines with Trifocal Tensors," *Multimedia Systems*, vol. 22, no. 2, pp. 181-195, 2016.

[7] Siam M. And Elhelw M., "Robust Autonomous Visual Detection and Tracking of Moving Targets in UAV Imagery," *in Proceedings of 11thIEEE International Conference on Signal Processing*, Beijing, pp. 1060-1066, 2012.

[8] Ristic S., "Flow Visualization Techniques in Wind Tunnels Part I, Non Optical Methods," *Scientific Technical Review*, vol.1, no. 1, pp. 39-50, 2007.

[9] Schreyer A. and Warfield S., *3D Image Processing*, Springer, 2002.

[10] Tunay T., Firat E., and Sahin B., "Experimental Investigation of the Flow Around a Simplified Ground Vehicle Under Effects of the Steady Crosswind," *International Journal of Heat and Fluid Flow*, vol. 71, pp. 137-152, 2018.

[11] Zhou S., Chellappa R., and Moghaddam B., "Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters," *IEEE Transactions on Image Processing*, vol. 13, no. 11, pp. 1491-1506, 2004.

Ahmad Drak is a researcher and PhD candidate at Bonn-Rhein-Sieg University o.a.s. Ahmad receivedhis B.Sc in Electrical Engineering from UAE University followed by M.Sc in Autonomous Systems from Bonn-Rhein-Sieg University. He has been working with robotics and UAVs in a research capacity for the last six years in topics related to fault diagnosis, control systems and sensor fusion. His areas of interest and research include multi-sensor fusion, object detection and avoidance, autonomous navigation and operation of robots and embedded systems. Currently he is working on his doctoral research which is concerned with efficient deployment and control of robots for maximizing information gain in dynamic environments.

Alexander Asteroth did his diploma studies in Mathematics and Computer Science at University of Bonn. Forhis PhD studies he received the Reihard Furrer scholarship of Wernher von Braun Stiftung zur Förderungder Weltraumwissenschaften. His research was done in cooperation with German Space Agency (DLR) and German National Research Center of Computer Science (GMD). His Research focus is on numerical optimization in various fields of application, usually involving surrogate models. He is currently a professor of Computer Science at Bonn-Rhein-Sieg University o.a.s and director of the Technical Institute of Resource and Energy-efficient Engineering.