Unmanned Aircraft Systems in Maritime Operations: Challenges addressed in the scope of the SEAGULL project

Mario Monteiro Marques, Pedro Dias,
Nuno Pessanha Santos & Vitor Lobo
Centro de Investigação Naval (CINAV)
Marinha de Guerra Portuguesa (MGP)
Almada, Portugal

Maria de Fátima Nunes, Elói Pereira
& José Morgado
Centro de Investigação da Academia da Força Aérea
Força Aérea Portuguesa (FAP)
Sintra, Portugal

Ricardo Batista & D. Salgueiro
Critical Software, S.A.
Lisboa, Portugal

Ricardo Ribeiro, Jorge S. Marques,
Alexandre Bernardino, Miguel Griné & Matteo Taiana
Instituto de Sistemas e Robótica (ISR)
Instituto Superior Técnico (IST)
Lisboa, Portugal

A. Aguiar, M. Costa, J. Estrela da Silva,
A. Sérgio Ferreira & J. Sousa
Faculdade de Engenharia da Universidade do Porto (FEUP)
Porto, Portugal

Abstract –The SEAGULL project aims at the development of intelligent systems to support maritime situation awareness based on unmanned aerial vehicles. It proposes to create an intelligent maritime surveillance system by equipping unmanned aerial vehicles (UAVs) with different types of optical sensors. Optical sensors such as cameras (visible, infrared, multi and hyper spectral) can contribute significantly to the generation of situational awareness of maritime events such as (i) detection and georeferencing of oil spills or hazardous and noxious substances; (ii) tracking systems (e.g. vessels, shipwrecked, lifeboat, debris, etc.); (iii) recognizing behavioral patterns (e.g. vessels rendezvous, high-speed vessels, atypical patterns of navigation, etc.); and (iv) monitoring parameters and indicators of good environmental status. On-board transponders will be used for collision detection and avoidance mechanism (sense and avoid).
This paper describes the core of the research and development work done during the first 2 years of the project with particular emphasis on the following topics: system architecture, automatic detection of sea vessels by vision sensors and custom designed computer vision algorithms; and a sense and avoid system developed in the theoretical framework of zero-sum differential games.

Keywords—Computer Vision; Unmanned Aerial Vehicles; Sense and Avoid.

I. Introduction

Portugal is well known as a maritime country with exceptional features for the incubation of technology that can be applied in the maritime field. It has the 20th largest Exclusive Economic Zone (EEZ) in the world and the 5th largest amongst the European countries (Fig. 1). The continental shelf, in accordance with the proposal submitted to the United Nations, may be extended to more than 2 million square kilometers. Portugal is also responsible for the 15th largest SAR (Search and Rescue) area in the world and the 2nd largest in the North Atlantic. This area is 62 times the national terrestrial space.

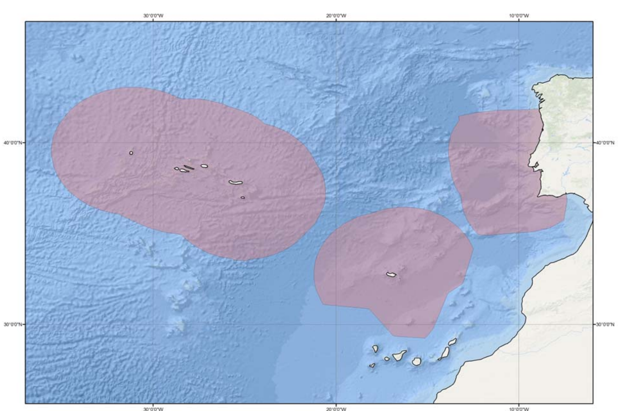

Fig. 1: Current EEZ and continental platform extension.

Nowadays, the maritime landscape perception is typically based on coastal radars, vessel-positioning systems, alarm systems, remote sensing and sensors installed in sub-surface, surface and manned aerial resources. However, there is still a huge gap regarding the observation capability, particularly in areas farther from the coast, where it is fundamental to increase the coverage, temporal resolution and spatial resolution of available means.

Moreover, the engagement of manned resources into patrolling this entire area constitutes a problem that translates into high costs (both in acquisition and in their operation), deficient coverage and poor temporal and spatial resolution of data collection to generate maritime situational awareness.

In this scenario, the use of unmanned autonomous vehicles, particularly aircrafts, with high autonomy, intelligence and reduced needs of communication and human resources appears as a very promising, perhaps unavoidable, alternative.

Therefore, the project aims to develop efficient solutions to address the challenges of maritime situational knowledge management. This perception is essential in maritime operations related to the safety of life at sea, security and environmental protection.

This project intends to develop intelligent systems that can be integrated in current unmanned aerial vehicle (UAV) technology, providing such capabilities as: detecting, identifying and tracking targets (e.g. vessels, environmental stains and spills, shipwrecks, debris, etc.); recognition of behavioral and planning patterns; collaborative missions with other autonomous vehicles (sense and avoid); monitoring of environmental parameters.

The paper is organized as follows. In Section II, we present the system architecture and its main components. In Section III, we describe the automatic detection system focusing on recently developed computer vision algorithms for vessel detection capable of real time operation on the embedded computers. Section IV describes the sense and avoid system and the algorithms for collision avoidance based on zero-sum differential games. Finally in Section V we present a discussion of the current and future project developments.

II. System Overview

A. System Architecture

The system architecture is oriented towards the definition of an open architecture based system, maximizing the use of open source and open standard components, protocols and interfaces (both hardware and software). This approach maximizes the interoperability, flexibility and durability of the resulting system, and facilitates future roadmaps created from these prototypes.

Another main architectural requisite is to obtain a low level coupling between the various functions of the system. Therefore the SEAGULL system is composed by several functional modules: the main modules deal with the command and control functionalities (SEC2) and with the detection of objects through the cameras and sensors aboard the aircraft (SEP), as illustrated in Fig. 2.

This modular approach not only supports the specific implementation of this project, but also guarantees that future developments such as new functionalities or new sensors can be integrated with minimum effort.

B. Robotic System

The robotic system aboard the UAV is composed by three main systems.

The Piccolo microcontroller is a Commercial Off-The-Shelf (COTS) device that implements the aircraft autopilot and directly actuates the UAV control surfaces [1]. It is connected to a Differential GPS (DGPS) for more accurate positioning information, allowing precision and autonomous landing. Piccolo has internal GPS, inertial and air data sensors that provide navigation telemetry to other systems and a data link radio that manages the communications channel with the command and control ground station (ETC2). This station then communicates with the payload ground station (ETP), which provides information to the maritime operations support and information system (SISOM). The connection between the Piccolo microcontroller and SEC2 is established through a serial port, while the connection with ETC2 will be through RF communication.

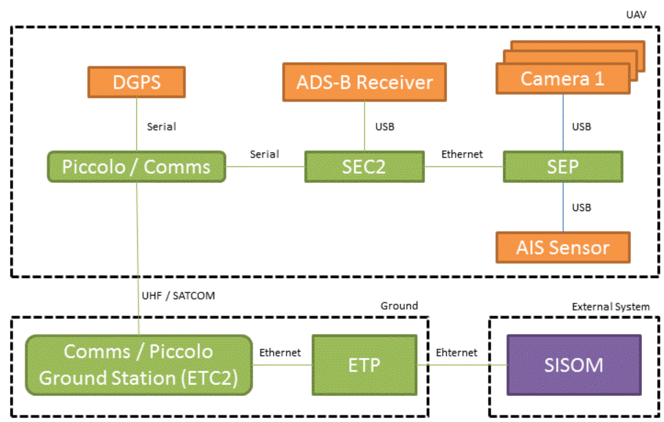

Fig. 2: Physical view of the system.

The computer of the command and control embedded system (SEC2) connects all SEAGULL systems to Piccolo, including maneuvering control (for sense and avoid and target tracking) and communications. Since this computer has direct connection to Piccolo, the Automatic Dependent Surveillance – Broadcast (ADS-B) receiver is connected to it providing traffic advisory and self-separation to other aircrafts and the DGPS allows precision and autonomous landing.

The computer of the Payload embedded system (SEP) contains all of the target detection functionalities, being connected to all the sensors used towards that goal. SEP is also connected to SEC2 in order to receive all navigation telemetry data that comes from Piccolo. This connection is also necessary to guarantee that SEP can activate/deactivate the target tracking mechanism and send/receive data to/from the ground station. Due to the computational needs derived from the image processing algorithms, this computer has higher performance requirements than the other systems and a GPU compatible with OpenCL.

The UAV is also equipped with an Automatic Identification System (AIS) receiver that fetches data broadcasted by vessels, e.g. Maritime Mobility Service Identity (MMSI), flag and GPS location. In order to detect vessels without AIS identification, the UAV is equipped with a computing vision system comprised by three cameras operating at different spectra – thermal, near infrared, and visible – and the onboard computer responsible for image acquisition and processing (SEP). The detected targets are georeferenced and provided to SEC2 onboard computer, which runs the target tracking controller and ensures that the target of interest remains in the center of the image frame [2].

A generator that draws power from the engine, backup batteries, and a power management board make up the power supply system. The power management board is responsible for turning on and off sensors during flight and has a built-in failsafe system for ensuring a non-disruptive power supply to the primary systems even in the event of a generator failure.

This project uses the ROS (Robot Operating System) middleware) [3]. This open source middleware supports asynchronous communication between processes, allowing them to publish messages into topics that can be subscribed by other processes with no limit of subscribers or publishers in any given topic. In addition to sending asynchronous messages, ROS also allows synchronous services, in the form of remote procedure calls (RPC). Another feature of ROS, which is highly advantageous in this project, is the ability to record data through rosbag package. Rosbag allows the user to record to a hard disk all the messages on any desired topic during the execution of a task, thus allowing a playback of the system's execution at a later stage, and providing a complete log for offline analysis after the UAV lands and completes its operation.

The UAVs used in the Seagull project (Fig. 3) were developed and manufactured at the Portuguese Air Force Research Center, and have a maximum take-off weight of 25 kg, a payload of 10 kg and an endurance of about 8h.

Fig. 3: Portuguese Air Force UAV.

III. AUTOMATIC DETECTION

One of the main topics of research in the SEAGULL project is the automatic detection of sea vessels by vision sensors and custom designed computer vision algorithms. The use of vision sensors is justified by its low cost, weight and energy consumption, which makes them well suited for small-to-medium scale UAVs. However, they pose some challenges due to sea surface image artifacts such as sun glare, wakes, wave crests, cloud shadows and debris, that can reduce the performance of the algorithms. Furthermore, the sea surface presents very diverse aspects depending on the atmospheric conditions and lighting variations during the day and year.

To design and evaluate computer vision algorithms able to cope with this variability, it is necessary to collect data and create datasets of video sequences characterizing this variability and containing samples of the objects and events of interest. We are creating a large database, with 7 sequences acquired from a GoPro camera at different conditions, with thousands of labeled images. This dataset is being used to develop and evaluate computer vision algorithms supporting the considered missions. Fig. 4 shows a few samples of images in the dataset, illustrating the diversity of scenarios.

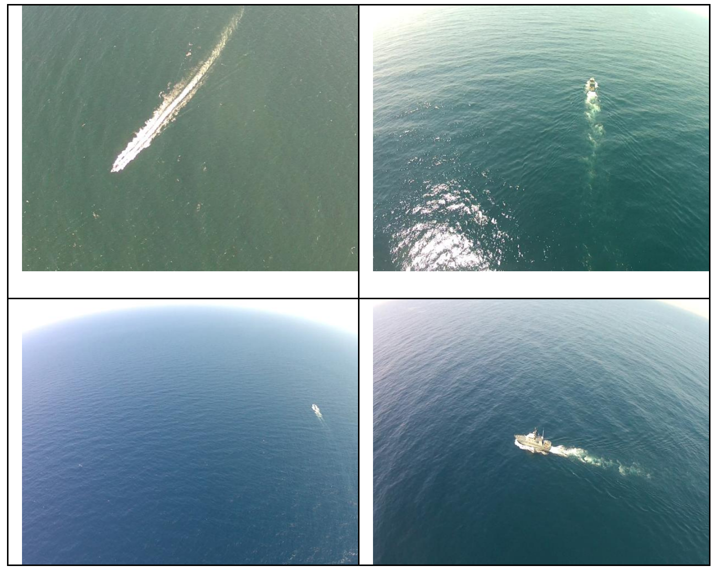

Fig. 4: Sample images from the labeled dataset of aerial images for the development and evaluation of vessel detection and tracking algorithms, illustrating some of the challenging conditions of the dataset: sun reflections, wakes, scale changes.

In Table I we describe the characteristics of the currently labeled data sequences.

TABLE I. VIDEO SEQUENCES IN THE LABELED DATASET.

ID	#frames	Description
1	1275	Small vessel; Clean conditions
2	2250	Medium vessel; Sun reflections; Small boat appears
3	505	Small vessel; Clean conditions
4	1070	Medium vessel; Sun reflections; Scale changes
5	1040	Large vessel; Sun reflections; Scale changes
6	749	Medium vessel; Sun reflections; Wake
7	2249	Medium vessel; Sun reflections;

The labeling of a dataset for detection and recognition is often a very burdensome task. It consists in tagging all objects in the image with a rectangular bounding box indicating the target location and size. Such operation needs to be done manually because it will be taken as ground truth against which the results of the algorithms are evaluated. Any kind of error in the labeling process will have a negative impact in the performance benchmarks and will compromise the reliability of the results. This manual process is quite heavy and long, especially if the number of images is large, as normally is the case for video sequences.

For that reason, a manual labeling tool with a light and fast interface was developed to facilitate the labeler's task. Several methods were implemented to minimize the number of actions needed to perform the labeling. A particularly useful one is an automatic labeling procedure for video sequences that adjusts the bounding box on the target, given initial (A) and final (B) positions (Fig. 5). It consists of a linear interpolation of the trajectory between point A and B combined with a local correlation search for fine adjustments [4]. If the trajectory of the target is approximately linear between the initial and end points, this method is able to compensate for residual camera and object movements. The user is then able to verify the labels obtained, but in most cases the labels obtained automatically are either correct or require only small adjustments.

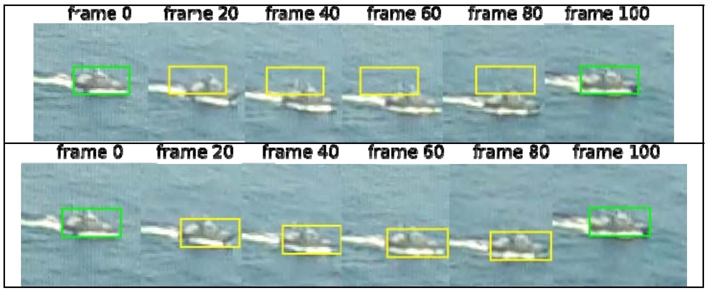

Fig. 5: Automatic labeling tool for video sequences. The user selects two frames of the video, between which the target trajectory is approximately linear. The first step (top) consists in linearly interpolating the target trajectory. Then, a local image correlation method (bottom) adjusts the location of the bounding box to the actual target location.

A few algorithms have been developed for the automatic detection of vessels from aerial images. Among the developed algorithms, the one described in [5] has shown the best results in comparison to the state-of-the-art, both in computational efficiency and precision-recall statistics. The developed method is based on blob analysis techniques enriched with spatial rules to reject sky and specular regions, followed by temporal filtering to reject blobs that are not consistent. In Fig. 6 we show some examples of detections obtained by our algorithm in different conditions of illumination and target size and posture. We can observe significant sun reflections, together with scale and rotation variations of the vessel, but the method is able to cope with these aspects, as long as the vessel is not inside the sun reflection area.

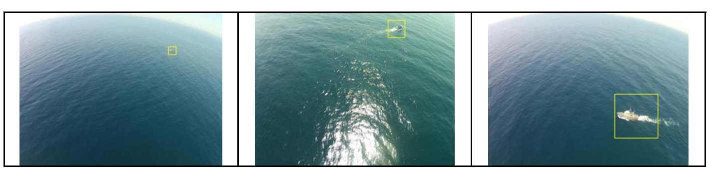

Fig. 6: Results of detections obtained with our algorithm, showing its robustness to rotation and scale variations, as well as specular reflections due to sunlight.

The proposed method is able to run in real time on the embedded UAV computational boards and performs faster and better than many machine learning based methods in many of the acquired video sequences. We have performed a quantitative evaluation of our method [5] with a state-of-the-art machine learning algorithm [6]. The method in [6] is based on rotationally invariant histograms of oriented gradients (HOG) [7] and trained with support vector machines (SVM) [8]. To improve it both in terms of computational complexity and precision, we pre-process the images using a saliency method to reject image regions with low contrast, and a sun reflection filter to reject large saturated regions in the image. Details of the implementation are presented in [9]. We have considered the single frame detection problem, so the temporal filtering step was removed from [5]. The method from [6] was trained with clips 1 and 2 (small and medium vessel). Evaluation was performed in clip 5 (the one presenting large scale variations). In Fig. 7 we show the precision-recall plot of the methods. For the machine learning based method we show the curve generated by varying the algorithms confidence level. For our method, because confidence levels are not computed, we show the operating point obtained with a setting of parameters using cross-validation.

The proposed method roughly achieves a precision of 50% for a recall of 90%. Such performance cannot be achieved by any of the confidence values in [6]. We note that no temporal rules were applied in this case. When temporal rules are used, our method's performance achieves about 99% precision for 75% recall in this sequence [5]. The fact that our method is based on blob analysis and rule based spatial analysis makes it easier than machine learning based methods to achieve robustness to scale changes and other appearance variations.

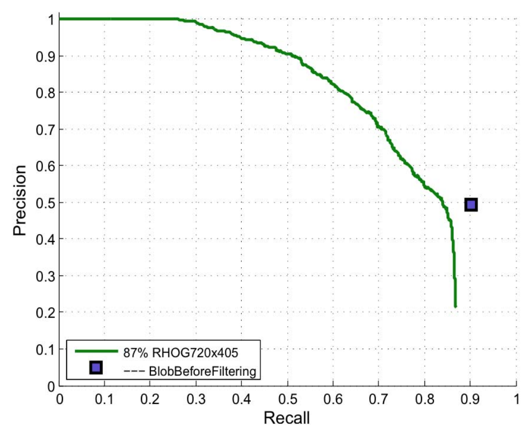

Fig. 7: Results of vessel detection algorithms in terms of precision-recall tradeoff. In blue we show the operating point of our method. In green we show the curve obtained by varying the confidence value in the algorithm of [2]. The performance of our method results in a better tradeoff between precision and recall.

Similar results were obtained in other sequences. However, in sequences 6 and 7, because the vessel is inside the sun reflection area during long periods of time and there is a significant amount of wave crests, the levels of performance diminished. These issues are being subject to further research and a more thorough analysis of results is being undertaken.

IV. Sense and Avoid

In order to ensure the UAS capability to operate safely in civil airspace, taking into account not only manned aircraft as well as other UAVs, it is necessary to provide the vehicle with the ability to detect other elements with which it shares its flight area [10]. This capability is increasingly important as the UAVs are proliferating and its use is increasingly growing not only in military as well as civilian applications [11].

It is generally accepted that the widespread use of UAVs will not be possible unless this ability is available. According to Schaeffer et al, in the US there are approximately 0.5 midair collisions per 1 million flight hours. Most of these occur near an airport within the uncontrolled airspace [12].

In order to address this issue we employ a transponder location system, with reception and transmission ability, allowing the implementation of control algorithms that take advantage of the information received to safeguard UAV's navigation. This will allow a natural extension of operational scenarios in which the UAV will be able to participate successfully, especially in mixed-initiative areas [13, 14].

This section describes the development of a prototype of a controller for preventing collisions between an aircraft of fixed-wing and other objects flying at the same altitude. The operation of the aircraft is made at constant altitude at which it is assumed that any obstacle avoidance will be made by changing the orientation of the aircraft.

The controller will be integrated in an aircraft with the ability to detect the position of other aircraft in their vicinity. The controller calculates an orientation reference for the aircraft as soon as it detects other aircraft in a state involving the risk of collision.

The aircraft movement is characterized by the following model:

$$\dot{n} = V\cos(\psi) \tag{1}$$

$$\dot{e} = V\sin(\psi) \tag{2}$$

$$\dot{\psi} = \omega \tag{3}$$

where the tuple ($\dot{n}$, $\dot{e}$) defines the aircraft's position in the North-East reference, ψ is the orientation and ω is the angular velocity commands. This is the abstract model of the aircraft curving procedure, which is usually done by varying the bank angle. The limit values for the angular velocity command are defined based on the aircraft specifications.

The controller is designed based on a dynamic programming technique, more specifically the topic of differential zero-sum games with two players [15]. The theory of differential games extends game theory (static) for problems involving dynamic systems. In the case of zero-sum differential games with two players, the profit of one of the players (e.g., an aircraft in flight) will be the subject of another player (e.g., an aircraft in hot pursuit). Dynamic programming is a technique that lends itself to numerical solution of this type and problems.

In the context of differential games, the problem of collision avoidance is addressed as follows. A ban area is set around the aircraft based on operating practical considerations (e.g., legislation, aerodynamic phenomena). This prohibition area is set to the same resolution of the chosen grid for the numerical synthesis of the controller. Moreover, we consider the following assumption: the absolute value of the speed obstacles can vary from 0 m/s to the speed value of the currently controlled aircraft; the direction of the velocity vector can vary instantaneously.

We consider a fixed referential to the body of the aircraft, with the x-axis aligned along the aircraft (positive toward the front of the aircraft), the y axis aligned transversely to the aircraft (positive toward the right of the aircraft) and the origin in the center mass of the aircraft. We define V_1 as the obstacle's absolute speed, ψ_r as the direction of its movement relative to the fixed referential in the controlled aircraft and (xr,yr) as its position relative to the same referential.

Thus, the system model becomes:

$$\dot{x}_r = -V + V_1\cos(\psi_r) + \omega y_r \tag{4}$$

$$\dot{y}_r = V_1\sin(\psi_r) - \omega x_r \tag{5}$$

For notational convenience, we also define F(x, ω ,(V_1, ψ_r)) as the system state after a sampling period Δ, from an initial position x≡(xr,yr), and with constant inputs ω and (V_1, ψ_r).

The air space state, in the fixed referential, will be classified according to the following categories:

- **Exclusion Zone** - as defined above, if the obstacle reaches this zone the controller considers that a collision is imminent;
- **Danger Zone -** An obstacle that actively pursues the aircraft from this area would always be able to reach the Exclusion Zone therefore the aircraft must avoid obstacles entering this area. Still in case any obstacle emerges in this area, the controller is still able to choose actions that maximize the probability of avoiding such obstacle;
- **Alert Zone –** In this zone there is no danger of collision if the aircraft employs the action indicated by the controller;
- **Safe Zone –** In this zone the aircraft can choose any action.

This classification is made beforehand based on the evaluation of the results of a set of differential games. Initially, it is assumed that the obstacle actively seeks out the controlled aircraft. The result of each game is calculated according to the relative position of the obstacle. For each position of the obstacle, the differential game can have the following results:

- **Defeat -** for the controlled aircraft, if there is a situation where it is inevitable that the obstacle enters the Exclusion Zone;
- **Victory -** for the controlled aircraft, if there is no such strategy.

In the context of dynamic programming, we set a value function V(x), which will take an infinite value if victory is achieved by controlled aircraft and a finite value otherwise. The finite value corresponds to the time required for the obstacle and the controlled aircraft to impact each other, assuming that both use optimal strategies. In the states corresponding to the exclusion zone, the value function is set to 0. For the remaining states, the function value is calculated according to the following equation (applying the principle of dynamic programming):

$$V(x) = \Delta + \max_{\omega} \min_{(V_1,\psi_r)} V(F(x,\omega,(\mathrm{V}_1,\psi_r))) \quad (6)$$

The value is calculated according to the algorithm described in [16, 17]. In turn, the control algorithm is built based on the function value. For this purpose we define the following equation:

$$\omega(x) = \arg\max_{\omega} \min_{(V_1,\psi_r)} V(F(x,\omega,(\mathrm{V}_1,\psi_r))) \quad (7)$$

The function $\omega(x)$ is counter-field all the possible combinations of angular velocity references for the controlled aircraft.

All states for which $\omega(x)$ includes all possible references are considered safe. This means that in these states, the aircraft can use any angular velocity reference.

This system was already validated in simulations (see Fig. 8) and in real environments.

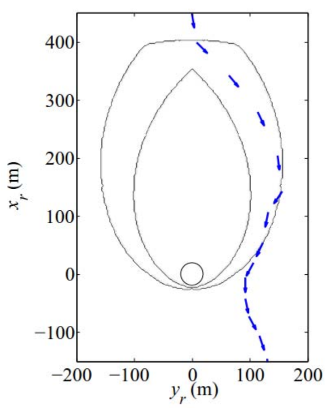

Fig. 8: Sense & avoid control algorithm. From the inner (circle) to the outer set: exclusion, danger and alert zones. In this simulation, the obstacle is following a straight line path, in route of frontal collision with the controlled aircraft (CA). The blue arrows show the relative path of the obstacle, with respect to the CA, while the evasive actions are being taken. As expected, the obstacle does not even enter the danger zone.

V. Conclusions

The overall objective of this work was the development of an intelligent system that can be coupled to unmanned aerial vehicles providing them with capabilities such as detecting, identifying and tracking targets; recognition of behavior and planning standards; collaborative missions with other autonomous vehicles; and monitoring of environmental parameters.

After several tests, both with real data and in simulation environments, it is safe to assume that this project is well aimed to meet the objectives for which it was proposed, producing a prototype that demonstrates the utility of UAVs in patrolling the Portuguese exclusive economic zone. With a well-established and tested platform, several algorithms for target detection, and a collision and avoidance mechanism, these vehicles have the required capabilities to be assigned to three types of missions – search and rescue, maritime surveillance and environmental monitoring – thus providing an increase in maritime situational awareness by the competent authorities and reducing the operating costs of such missions.

As future work it should be pursued a path to achieve greater autonomy, increased decision-making capacity by the on-board system and an improved range and bandwidth of the communication channel.

Acknowledgment

This work was funded by POFC (Programa Operacional Factores de Competitividade) within the National Strategic Reference Framework (QREN) under grant agreement 2013/034063(SEAGULL, project number 34063).

References

[1] Cloud Cap Technology Piccolo II manual, UTC Aerospace Systems, 2014

[2] T. Oliveira and P. Encarnacao, “Ground target tracking control system for unmanned and aerial vehicles”, Journal of Intelligent and Robotic Systems, vol. 69, pp 373 – 387, 2013 (oral presentation at ICUAS 2012 – 2012 International Conference on Unmanned Aircraft Systems, Philadelphia, PA, USA)

[3] Robot Operating System homepage, 2015, available from: <http://wiki.ros.org/>

[4] Trucco, E. & Verri, A., Introductory Techniques for 3-D Computer Vision, Prentice Hall 1998, pp. 146-148

[5] An Algorithm for the Detection of Vessels in Aerial Images. J. Marques, A. Bernardino, G. Cruz, M. Bento, AVSS 2014

[6] Rotation-Invariant HOG Descriptors Using Fourier Analysis in Polar and Spherical Coordinates. Kun Liu, Henrik Skibbe, Thorsten Schmidt, Thomas Blein, Klaus Palme, Thomas Brox, Olaf Ronneberger, Intl. Journal of Computer Vision, 106(3):342:364, 2014.

[7] Histograms of oriented gradients for human detection, N. Dalal and B. Triggs, CVPR 2005.

[8] Support-vector networks, C. Cortes and V. Vapnik, Machine Learning 20(3):273-297, 1995.

[9] Vessel Detection in Oceanographic Airborne Imagery, Miguel Griné, MSc Thesis, Instituto Superior Técnico, Universidade de Lisboa, 2014.

[10] Geyer, CM, Singh, S., & Chamberlain, LJ (2008). Avoiding Collisions Between Aircraft: State of the Art and Requirements for UAVs operating in Civilian Airspace (No. CMU-RI-TR-08-03). Pittsburgh

[11] Bencatel, R., Faied, M., Sousa, J., & Girard, AR (2011). Formation control with collision avoidance. In IEEE Conference on Decision and Control and European Control Conference (pp. 591-596). IEEE. doi: 10.1109 / CDC.2011.6160637

[12] Hoffmann, M. G., & Tomlin, C. J. (2008). Decentralized cooperative collision avoidance acceleration is constrained vehicles. In 2008 47th IEEE Conference on Decision and Control (pp. 4357-4363). Cancun: IEEE. doi: 10.1109 / CDC.2008.4739434

[13] Tomlin C. Pappas, G., & Sastry, S. (1998). Conflict resolution for air traffic management: a study in multiagent hybrid systems. IEEE Transactions on Automatic Control, 43 (4), 509-521. doi: 10.1109 / 9.664154

[14]]Roussos, GP, Chaloulos, G., Kyriakopoulos, KJ, & Lygeros, J. (2008). Control of multiple non-holonomic air vehicles under uncertainty using wind Model Predictive Control and decentralized navigation functions. In 2008 47th IEEE Conference on Decision and Control (pp. 1225-1230). Cancun: IEEE. doi: 10.1109 / CDC.2008.4738792

[15] Bellman, R., Dynamic programming. Princeton University Press. Princeton 1957

[16] Jorge Star Silva, João Borges de Sousa, On the numerical synthesis of optimal feedback controllers, 5th International Scientific Conference on Physics and Control, Leon, Spain, 5-8 September 2011

[17] Jorge Star Silva, João Borges de Sousa and Fernando Lobo Pereira, Dynamic programming based feedback control for systems with switching costs 2012 IEEE Multi-Conference on Systems and Control, Dubrovnik, Croatia, 3-5 October 2012