A Model for landing, taking off and autonomous battery recharging of a Parrot Ar.Drone 2.0 using computational vision and GPS features

José Felippe Loureiro[1], Robison Cris Brito[1], Carlos Eduardo Magrin[2], Eduardo Todt[2]

[1] Federal University of Technology - Paraná, Pato Branco, Brazil

[2]Federal University of Parana - Curitiba - Brazil

zefelippe_0@hotmail.com, (robison.brito,cmagrin,eduardo.todt)@gmail.com

Abstract. *A drone is a type of Unmanned Aerial Vehicles (UAV) that most of the times can have four propellers. They can be used in many applications, one of those is to move through places of difficult access. Besides drones practicity over other aerial vehicles, its price is way lower compared to large vehicles, which turns them attractive to many activities. Also it offers safety in dangerous situations, like fires or accidents, as it doesnt need an on-board pilot. In a system with autonomous flight the concern with its landing and recharging of the batteries, which does not last more than a few minutes, arises. Using on-board devices, like its cameras and GPS modules, it is possible to implement functions to optimize its capabilities. With the goal to present a solution to such problem, this essay proposes a model which utilizes image recognition to allow a drone to land in an autonomous system. This landing routine based on its image turns flight and landing into autonomous processes, without human intervention.*

1. Introduction

With the technological advances in electronics, unmanned aerial vehicles (UAVs) have become increasingly accessible, either due to their decreasingly cost and for the options available in the market.

The US research group Teal estimates that world production of unmanned aerial vehicles will increase from U$ 2.6 billion in 2016 to U$ 10.9 billion in 2025, with an annual growth rate of 15.4 %. Over the next decade the market will amount to a total of U$ 65 billion [FINNEGAN 2013].

In the civilian scope of application, some of the existing demands are environmental monitoring, such as pollution, climate or scientific applications, forest fires, border control, drug traffic, aerial surveillance, mapping, traffic monitoring, precision agriculture, search and rescue, among others [Bastianelli et al. 2012]. There is still a very large demand for the use of UAVs in the military, either for reconnaissance, surveillance, damage assessment after an attack, or as a contingency mechanism and for communication.

Among the types of UAVs are the drones, also known as quadcopters, which are usually smaller and, most of the time, simpler aerial vehicles. According to [Bastos 2015], drones have gained space, among other areas, in precision farming and livestock farming. Their versatility is worth the investment, since they can perform diverse functions in agricultural regions of difficult access and they have a relatively low cost.

Regarding the technological resources present in the drones, practically all current commercial models have video and / or photo cameras, ultrasound sensors to measure distance from the ground, GPS module for self-localization and modules for remote communication using radio-frequency, however, one can find more sophisticated drones with a greater number of sensors and features. As an example of drone with the basic resources cited is the AR.Drone 2.0 from the French company Parrot, with an approximate cost of U$350.

Although very useful, a drone is an equipment that needs to be handled with care, a fact that made it difficult for the homologation for commercial use by the Brazilian Air Force [ANAC 2017]. This is due to its rotating propellers and their weight, which can vary from a few grams to several kilograms (for homologation, the drones were divided into three categories: up to 2.5 kg, from 2.5 kg to 25 kg and above 25 kg). A drone can cause serious injury in the event of an accident or fall. In all categories, by homologation, autonomous flight can only occur in non-populated areas.

To minimize the risks, the National Civil Aviation Agency (ANAC) published in late 2017 the new regulation on commercial use of drones, which promises to boost and bring greater security to its commercial use in Brazil. The Brazilian regulations follow the line of action adopted by the International Civil Aviation Organization (ICAO), based on amendments to the Annexes of the Chicago Convention. The regulation that addresses the use of Brazilian airspace by remotely piloted aircraft must be constantly revised and adjusted, given the dynamic nature of the activity and the recurrent technological advances [ANAC 2017].

The major disadvantage of using drones is still the limitation of autonomy in battery vehicles. Typically quadcopter drones have a flight autonomy of up to 30 minutes, however, low-cost drones such as the Ar.Drone from Parrot have just 10 minutes. In addition, after the battery has run out, the recharging process is slow and requires human supervision.

Aiming to present a feasible alternative to overcome this limitation, this work presents a model for the solution of two of the major limitations related to the autonomous flight of drones - their landing/takeoff and battery charging, as human intervention is now necessary for both. In order to test the model, a prototype of the recharge base was developed, as well as an adaptation in the Parrot AR.Drone 2.0 to allow an autonomous recharge, and a software was also developed to communicate with the AR.Drone 2.0 during the landing/recharging/take-off process.

2. Materials and Methods

For the development of this work, a systematic mapping was carried out on works that involved cooperation between vehicles, and part of this study selected works that envolve autonomous takingoffs/landing/recharging [Brito et al. 2017]. These are detailed in the sequence.

[Cheng et al. 2013] presented a cooperative approach between a UAV and an Unmanned Terrestrial Vehicle (UTV) based on monocular vision for UAV landing. Using tracing to mark a target on the UTV, the UAV can track and land autonomously on the moving UTV. Control loops based on PID controllers are employed to perform two levels of control: stabilization control and position control. The stabilization control is performed on an on-board micro controller board with the aid of an inertia measurement unit. The images captured by the on-board camera are transmitted to a central terrestrial control unit, which is present on a laptop via wireless channels. The relative position of the UAV for the land vehicle is estimated from the received images, not by GPS, so the estimated current position of the UTV is sent to the UAV via Wi-Fi. The proposed vision-based approach to detect and locate the target, as well as the height of the UAV, is robust when the UTV is easily. Practical experiments segmented from the show that the UAV can autonomously monitor the UTV and background perform landing on the moving target. This work deals exclusively with the autonomous landing, not addressing the autonomous reloading and take-off operation.

[Barresi and Allasia 2013] has explored the UAV drive within an airport. Using the UAV front camera video. GPS information is also used for the auto-location of the UAV. The GPS, however, introduces a systematic error in the system, especially in the indoor areas. The project developed by the authors presents a new approach in which the processing of the images allows a rectification of the GPS signal, making use of the automatic recognition of airport signs and marks. The take-off paths as well as the taxiway location signs are identified and correlated with the GPS information. Several approaches were made using Hough's transformation to find the runway for landing or takeoff without cooperation between position and optical systems. The goal of the work is to create a routine that automates landing and takeoff based on images to improve the accuracy of GPS in indoor areas, but it does not address the autonomous recharging.

In another project, [Cocchioni et al. 2014] addressed cooperation between unmanned devices by performing missions in indoor/outdoor areas. In this, the authors focus on the interaction between UTVs and UAVs to extend the autonomy of flight of a UAV by means of a landing platform. The UTV acts as a recharging station and hosts the UAV during the indoor/outdoor transition and vice versa. The platform was designed with the goal of achieving a robust landing. Synchronization and co-ordination of co-operation are managed by a control station. This station was developed using a software tool based on the integration of Stateflow, automatic generation of C code and ROS (Robot Operating System). All software components of UAV, UTV and the station itself were developed using ROS. The results show that the UAV was able to land on the UTV with high precision (< 5 cm for the x and y axes) thanks to a visual position estimation algorithm. The present study, however, did not address the autonomous recharge of the UAV.

Also in the vehicle interaction scenario, [Lee et al. 2014] presented a method of cooperation between two UAVs, one flying in high altitude and the other in low altitude, to achieve autonomous navigation and landing. In the autonomous landing based on computational vision, the accuracy of the GPS signal and the efficiency of the tracking algorithm and target detection affect the performance of the autonomous landing system. In this way, by using the comprehensive view and the high flexibility of an UAV at high altitude, it is possible to control an UAV at low altitude so that it can perform the landing procedures correctly. The flight controller can track the target and control the device in real time. This is possible through a high level control system using fuzzy logic and neural networks to calculate the positioning and perform maneuvers of the low altitude device. The present work does not present a model for autonomous recharging and takeoff.

[Maini and Sujit 2015] state that aerial surveillance and mapping are the main areas for developing applications that involve small UAVs, such as drones. When it comes to the mapping of large areas, for example, one can have missions that last a longer time. However, the duration of the mission is limited by the ability of the UAV battery. This requires reloading the UAV for the success and completion of the mission. Points of interest in the area to be mapped may not be accessible from a single recharge station and therefore multiple stations are required. In addition, recharging stations could not be placed anywhere in the region due to terrain restrictions. Finally, the routes of the UAVS missions should be planned according to the layout of the refueling stations. So UAVS route planning and the provision of recharging stations are connected problems, and getting optimal solutions becomes difficult. The authors developed a greedy strategy of coordination between a UTV and the UAV, with the help of a simulation platform created with the help of mathematical software Matlab to test and validate the strategy. Field tests were conducted using a single drone. The proposed model predicts the landing/reloading/autonomous takeoff, however, they were only tested with simulations.

[Djapic et al. 2015] have been responsible for the Heterogeneous Autonomous Mobile Maritime Expeditionary Robots (HAMMER) project, which aims to integrate an autonomous surface platform with three different types of unmanned vehicles: aerial, surface and submarine. The HAMMER system consists of several marine vehicles working together, among them the unmanned surface vehicle (USV) acts as the central node and main transport mechanism, it can be used to transport UAVs and unmanned underwater vehicles (UUV). The system is designed to be modular and can be easily extended. For the implementation of the USV, a 16-foot catamaran (70x40-in landing area) was used. By using the surface vehicle for the base of operations of a UAV, it is possible to guarantee the interoperability, coordination and cooperation of autonomous mobile marine robots in environments in which access to the GPS is denied. The three main research areas related to the project are image processing, state estimation and autonomous cooperative control. Due to the challenges of the maritime environment, both software and hardware used must to provide the level of flexibility and resources required to achieve an efficient and robust landing. The communication should also be reliable, being developed to allow the efficient sharing of data and control messages between the surface and the aerial platforms. Although it deals with landing and take-off autonomous, the work does not deal with recharging the battery.

All the works presented in this section use some form of landing and/or autonomous take-off of an UAV. All have their merits, but the recharge action, given the technical limitations, were not addressed.

The model proposed by this work aims to automate besides the landing and take-off, the autonomous recharge of the battery, being this model implemented on a recharge base for a Parrot AR.Drone 2.0 using GPS and computer vision features.

3. Results and Discussion

The aim of the work is to propose a model for landing/take-off and autonomous reloading of an Parrot AR.Drone 2.0. Its approach to the base is accomplished through the aid of a GPS and its landing happens with the aid of image processing on the images acquired by the drone's on-board camera. As soon as the drone connects to the base, the charging of the battery begins. Upon completion of the recharge, the drone is ready to take off and continue its mission. The model is shown in Fig. 1.a). Communication between the

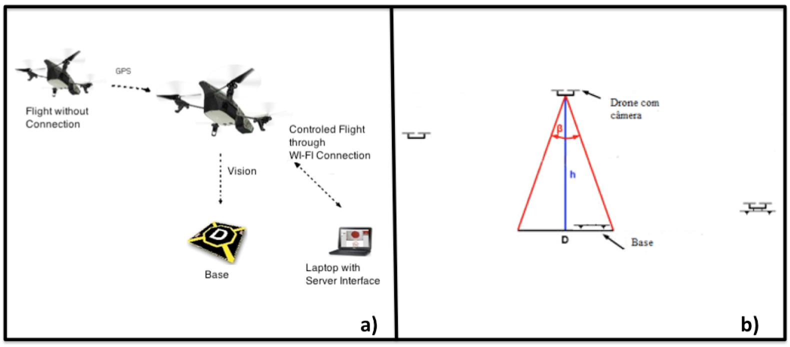

Figura 1. a) Landing/take-off and recharging model using GPS and drone image processing. b) Calculation of the control transition height, bellow this height control is transferred from the drone to the recharging base.

recharging base and the drone is done via a Wi-Fi connection. As the drone approaches the base, it establishes a Wi-Fi connection and transfers the images captured by the drone. The base is responsible for processing these images and sending maneuver commands to the drone, however, this happens only when the drone is in the process of landing, that is, near the base.

For the prototype, it was decided to use a drone that has stood out in the world market for its reduced price and ease of use. The Parrot AR.Drone 2.0 is equipped with a 32-bit processor, has Linux operating system and allows its control of the flight through a remote device (smartphone, computer, micro-controller, among others), and it must be connected to the drone using Wi-Fi. Its battery allows flights of 10 minutes in average. This drone can be equipped with the FreeFlight - GPS module for its self-location capability. For these reasons the AR.Drones are commonly used in scientific research, as presented in [Lioulemes et al. 2014], [Aitken et al. 2016] and [Duarte 2017].

The purpose of our job is to assist a drone in its landing when there is a need for autonomous charging, allowing the battery to charge and leaving it fit for subsequent take-off. For this purpose, the drone should, at the time it detects a low battery signal, find the geostationary coordinate of the nearest recharge base. This information will be present in the micro-controller of the drone, being processed by an embedded software, which constantly reads the battery level and when being lower than a threshold value, will cause the drone to fly to the recharging base.

Being near the base a Wi-Fi communication between the drone and the base is established, initiating the process of descent, which uses as a parameter the image recognition obtained the drone itself. This causes the drone to align with the base and make the landing. Charging occurs when the drone's feet touch the metallic contacts of the recharge base.

In order to identify the minimum height at which the drone needs to be for the correct recognition of the image of the recharge base, a trigonometric function was required, as shown in Fig. 1.b). Considering that the GPS module of the equipment has a precision of around 2 meters, that is, given a certain geostationary point, the drone will recognize it being 2 meters more or less from the point (D). The angle of vision of the lower camera is 63 degrees (β). Having these two information at hand and using trigonometry it's possible to figure out the height (h) that the drone must fly to catch the base in it's field of view, which is at least 3.5 meters. Thus, when the drone descends and reaches this height, its control is carried out from instructions sent from the base to the drone, which forwards instructions for an accurate landing on it, based on the images received from the drone itself.

For the landing of the drone a base is needed that is easily recognized and that facilitates the landing and recharging. In this way, the base was designed as a circumference, with signs to identify the direction the drone should land (the front of the drone is facing the corresponding side of the base). It was chosen a base in the circular format to aid the identification of this by means of algorithms of image processing, since geometric images are easily identified in this type of algorithm. In the same way, the red color was chosen to identify the position in which the drone should descend. This color was chosen to have a counterpoint to the land area, which is normally an agricultural area, the background is usually green when there is vegetation, or brown of exposed soil.

Geometric objects are relatively easy to identify due to their specific characteristics such as perimeter, area and radius. Even on the basis of the image, the precision required for the contact must be high. In this way the base has 4 holes in the shape of an inverted cone, allowing the drone to land even with a small positioning error, the contact is driven by gravity to the correct position for recharging. Thus, each foot of the AR. Drone will fit into one of these orifices of the inverted cones, as shown in Fig. 2.a). Each cone has 9 cm in radius and 12 cm in height.

The drone battery recharge is achieved by means of an adapter in the base that must be connected to a power source. The feet of the drone were fitted with metallic tips, connected to the three cells of the battery, so that at the moment of landing, they make contact with charger. The tip of the cone contains a metal base to ensure contact with the battery poles. The Fig. 2.b) demonstrates the adaptations.

The base was made using a MDF board, measuring 60 centimeters in circumference, in which were drilled four circular holes 9 centimeters in radius. These holes serve as accommodation for the 4 designed cones, which were manufactured with a 3D printer. The contact plates present on the inside of the cone were made using circular-shaped electrical plates and were welded to the cables connecting to the charger. The base and cones are shown in Fig. 2.c).

For reasons of compatibility with the AR.Drone control environment, the server runs on a notebook connected to the recharge base. The server software was developed in Node.JS and Angular.JS using the JavaScript language. These tools facilitate the development of activity due to AR.Drone-compatible libraries, available for Node.JS, which contains functions for drone handling. This server is connected to the recharge base and is responsible for processing the information (images) sent by the drone. Based on these, images commands and maneuvers are sent to the drones.

For the application, a web server is required for message exchanges and communication with the drone. This system boils down to a screen for displaying the image received from the drone, as well as the result image after recognition of the desired color, and a few buttons to control the drone. This graphical interface was used only for the tests, and interaction with the human being in the real application is not necessary. This appli-

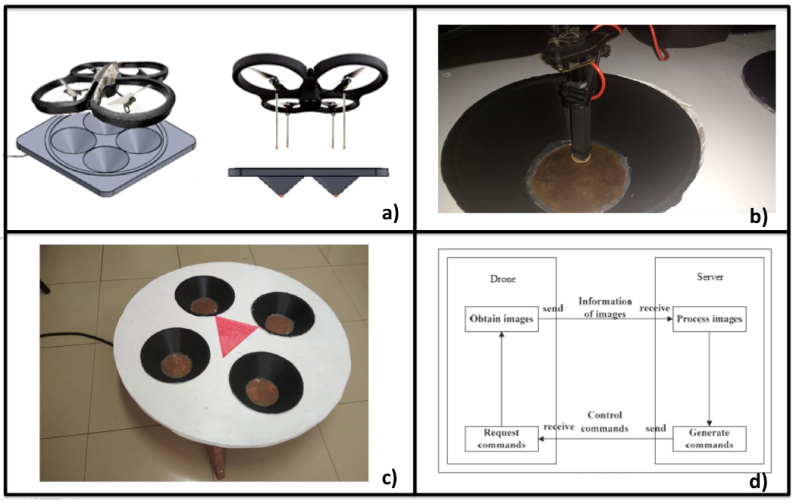

Figura 2. a) Representation of the drone approach and contact if the drone feet with metallic contacts at the base. b) Adaptation of the contacts on the drone feet with the metal contacts on the base. c) Prototype of the recharge base developed for landing and recharging of an ArDrone. d) Representation of communication between drone and server.

cation is also responsible for sending the flight commands, processing the drone image, identifying the drone's position to handle the landing, and sending this command via Wi-Fi as a response to the drone. The drone can be considered a Wi-Fi router, which accepts connections from various devices, in this case the recharge base. Fig. 2.d) demonstrates the communication between drone and base.

As Node.JS and Angular.JS have great compatibility with JSON and both use JavaScript for development, client and server were created in such a way that they communicate via JSON. The client and the access to the information were codified using the technologies HTML and Angular.JS, since they are languages widely used in the Internet and are compatible with each other. The drone captures its images in JPEG format and they are handled by the server using the OpenCV open library.

OpenCV is an open source library of computer vision and machine learning, which facilitates its use, as well as the modification of the code by users. The library has more than 2500 optimized algorithms, which includes a comprehensive set of computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, track moving objects, extract models of 3D objects, and more.

To identify the recharge base, an algorithm based on color detection and geometric figures was defined. Fig. 3.a) shows the flow of the algorithm.

Generally, it is assumed that the RGB color space is more suitable for color-based segmentation, however, the HSV color space is the most appropriate. The HSV color space consists of 3 matrices, wich are the matrix of hues, saturation and values. In OpenCV, the range of values for the fields of the matrices are, respectively, 0-179, 0-255 and 0-255. Hue represents the color, saturation represents the amount with which the respective color is mixed with white and value represents the amount with which the color is mixed with black (Szeliski, 2010). In this way, when the image is transformed into the HSV space, a color ends up being defined by a single parameter, instead of three as it is

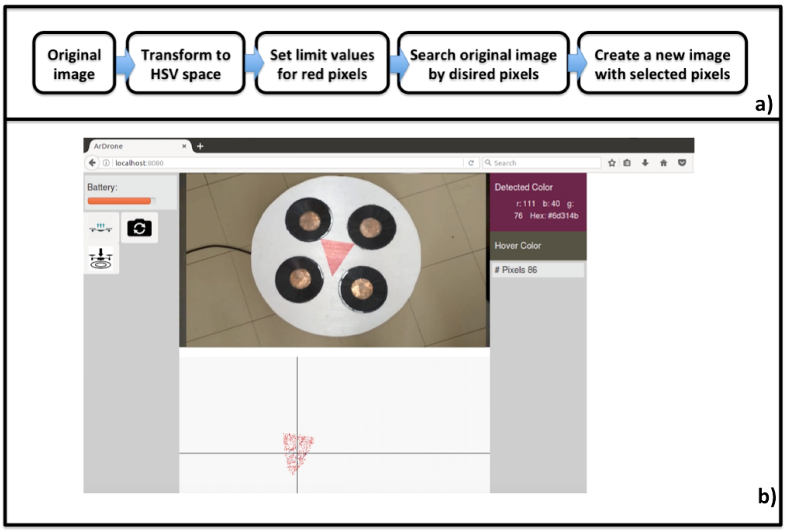

Figura 3. a) Algorithm for color detection and image processing. b) Application main screen.

in the RGB, thus facilitating the processing.

The red color was chosen to be used as reference color for the drone approaching the base, and has values between 170-180, 160-255, 60-255. Here, the hue is defined exclusively for this color distribution. Already saturation and value may vary depending on the ambient lighting condition. In this way it is necessary to perform two searches, one for the lower limit of the values and one for the upper limit. Finally, adding the two images we have a third image as a result, which contains only the desired pixels.

So it is possible to extract relevant information from the result image. Since the image contains only the pixels in which the red color was recognized, a function is used to discover the center of the present area. With this we have the values of X and Y necessary to control the centering of the drone in relation to the base.

The application consists of a single interface, containing a battery meter, three buttons, one to take off, one to land and the last to change the camera being used (since the AR.Drone has two cameras, one in the front and one in the bottom), two image fields, one displaying the original image received by the drone and the other the image already processed, and finally an interface that allows the choice of the color to be searched for. The Fig. 3.b) displays the application screen.

The algorithm presents satisfactory results, recognizing the red color among others tested. A major problem encountered is the intensity of illumination on the object in focus. Often a shadow or difference of illumination in an object causes the incorrect identification of colors that should be found. The image quality generated by the AR.Drone's lower camera is lower than the front camera and did not perform well. Even with the problem of illumination in uncontrolled environments, the correct identification of the base happened at about 80 % of the descents.

As the project progressed, some problems had to be addressed. According to the AR.Drone 2.0 User's Guide, the device must not have its battery charged while the device is on. This problem has been solved by making a specific circuit to cut off the drone's power at the moment of landing for a set time (40 minutes, time required to fully charge the drone). An opto-coupler drive circuit was designed to ensure insulation of the components of the drone's from the battery recharge, so that upon landing, a signal received from the base triggers the TRIAC and reverses its logic of action, cutting the energy of the micro-controller.

In this way, as the present model is projected, when an AR.Drone has the battery at critical level, it pauses the mission (the mission is present in the drone micro-controller), returns to the base of recharge, executes the landing procedure, stays recharging the battery for 40 minutes. After that the recharge base disconnects from the drone, which takes off and continues the mission from where it paused.

4. Conclusion

The present work developed a model for landing, recharging and autonomous take-off of a Parrot AR.Drone 2.0. In spite of the initial difficulties, after a series of studies and calibration, it was possible to have the drone approach the base, descend, recharge the battery autonomously, being fit for a new flight.

The use of the Node.JS language was of great value for the project, especially considering the integration of the drone with the recharge base, as well as for the development of a friendly visual interface for tests. As the language has a specific library to assist the control of the AR.Drone, it has facilitated the control of the drone. In the same way, the JavaScript language is fundamental to web application development. The project shows that it is possible to achieve the desired goals.

The drone used can be obtained online with prices starting at $ 300.00. The cost for the base and cones of the base was $ 8.00. You also need a computer for the application server (this can be replaced with a Raspberry Pi).

The project is feasible and with great potential for improvement. With the help of the OpenCV library it was possible to assemble several applications based on image recognition. The biggest problem is with the image quality of the lower camera of the drone, which is of low quality and often makes processing difficult.

As an alternative to future works, the lighting control of the base could be improved, increasing the drone's descent efficiency (a faster and more accurate recognition of the base), also tests with new colors and new base formats, as well as transforming the prototype into a more professional product, already embarking in the own drone the circuit that turns off the drone when it is in the base of recharge, avoiding any damages to the drone.

Referências

Aitken, J. M., McAree, O., and Veres, S. M. (2016). Symbiotic relationship between robots — a ROS ARDrone/YouBot library. In *2016 UKACC 11th International Conference on Control (CONTROL)*. IEEE.

ANAC (2017). Resolução n429, de 02 de maio de 2017. In *Requisitos Gerais para Aeronaves não tripuladas de uso civil*. Agência Nacional de Aviação Civil.

Barresi, F. F. and Allasia, W. (2013). Airport markings recognition for automatic taxiing. In *Conference: Design and Architectures for Signal and Image Processing (DASIP),*. Institute of Electrical and Electronics Engineers (IEEE).

Bastianelli, G., Salamon, D., Schisano, A., and Iacobacci, A. (2012). Agent-based simulation of collaborative unmanned satellite vehicles. In *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*. Institute of Electrical & Electronics Engineers (IEEE).

Bastos, T. R. (2015). 15 usos de drones na agricultura e na pecuária. In *http://revistagloborural.globo.com/Noticias/Pesquisa-e-Tecnologia/noticia/2015/05/15-usos-de-drones-na-agricultura-e-na-pecuaria.html*. Acessado em 23 de maio de 2018.

Brito, R. C., Loureiro, J. F., Todt, E., and Pereira, R. (2017). A systematic mapping for the scenario of non-urban autonomous vehicle cooperation systems. In *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*. IEEE.

Cheng, H., Chen, Y., Li, X., and Wing, S. W. (2013). Autonomous takeoff, tracking and landing of a uav on a moving ugv using onboard monocular vision. In *32nd Chinese Control Conference*. Institute of Electrical and Electronics Engineers (IEEE).

Cocchioni, F., Pierfelice, V., Benini, A., Mancini, A., Frontoni, E., Zingaretti, P., Ippoliti, G., and Longhi, S. (2014). Unmanned ground and aerial vehicles in extended range indoor and outdoor missions. In *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE.

Djapic, V., Prijic, C., and Bogartz, F. (2015). Autonomous takeoff & landing of small UAS from the USV. In *OCEANS 2015 - MTS/IEEE Washington*. IEEE.

Duarte, R. (2017). Low cost brain computer interface system for ar.drone control.

FINNEGAN, P. (2013). World unmanned aerial vehicle systems, market profile and forecast 2013. teal group research. In *http://tealgroup.com/index.php/about-teal-group-corporation/press-releases/129-teal-group-predicts-worldwide-civil-uas-production-will-total-65-billion-in-its-2016-uas-market-profile-and-forecast/*. Acessado em 23 de maio de 2018.

Lee, M.-F. R., Su, S.-F., Yeah, J.-W. E., Huang, H.-M., and Chen, J. (2014). Autonomous landing system for aerial mobile robot cooperation. In *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*. IEEE.

Lioulemes, A., Galatas, G., Metsis, V., Mariottini, G. L., and Makedon, F. (2014). Safety challenges in using AR.drone to collaborate with humans in indoor environments. In *Proceedings of the 7th International Conference on PErvasive Technologies Related to Assistive Environments - PETRA 14*. ACM Press.

Maini, P. and Sujit, P. B. (2015). On cooperation between a fuel constrained UAV and a refueling UGV for large scale mapping applications. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE.