An Autonomous Drone for Image-Based Inspection of Bookshelves

Ester Martinez-Martin
Dept. Computer Science and Artificial Intelligence
University of Alicante
San Vicente del Raspeig (Alicante), Spain
ester@ua.es

Eric Ferrer, Ilia Vasilev and Angel P. del Pobil
RobInLab, Dept. of Engineering and Computer Science,
Jaume I University
Castellón, Spain
{al312992,al365834,pobil}@uji.es

***Abstract*—This article introduces the UJI aerial library robot. It uses visual techniques to self-locate and navigate autonomously to find books and make automated inventories. A control strategy for navigating along library shelves is presented, using visual markers for self-positioning. An image-based book recognition technique is described, which combines computer vision techniques to detect the labels on the spines of the books, and then optical character recognition (OCR) to decode the book code into text. These data can be used to make an inventory of the library. Lost books can be detected automatically and a particular book can be located in the library. Our quadrotor robot was tested in a real library with promising results.**

***Index Terms*—drone, image analysis, localization, indoors, navigation, UAS, automated inventory, indoors, book recognition**

I. Introduction

A library is a suitable environment for robotics application since most of the librarian tasks are repetitive and time-consuming and they could be automated, such as inventory or relocation of misplaced books [1]. A considerable research has resulted in several technologies that try to automate these tasks, including librarian robots for inventory tasks [2], [3], [4]; computer-vision-based book recognition [5], [6], [7], [8]; and book detection based on deep learning [9], [10], [11].

With respect to book delivery, a pioneering system was the UJI librarian robot [12], an autonomous mobile manipulator for localising and extracting books from the bookshelves in a library desk. Also, RFID tags have been used [13].

On the other hand, aerial robotics has grown to be a popular field in the last decade to the extent that Unmanned Aerial Vehicles (UAVs) -or drones- have become a standard platform in the robotics research community [14] thanks to their versatility, high mobility and ability to cover areas at different altitudes and locations. Indeed, this kind of robots have enabled a large variety of applications, such as traffic monitoring, homeland security, farming, surveillance, etc. [15].

Compared with traditional wheeled robots, the main advantage of aerial robots is their ability to move in three-dimensional space with little effort, flying at different altitudes and hovering in the target area to collect precise information. This ability to move in 3D space, however, brings with it great scientific and technical challenges, specially in the case of autonomous flight in indoor spaces [15] for which perceptual intelligence based on aerial vision is called for to self-localise, navigate and perform the desired tasks in human environments. Few such indoor systems exist; for instance, Harik et al. [16] describe an approach combining an Unmanned Ground Vehicle (UGV) and an UAV for the automation of warehouse inventory.

An additional problem is that Global Navigation Satellite System (GNSS) cannot work properly indoors; as an alternative, computer vision technologies have been used for navigation of UAVs [17]. The use of maps is a state-of-the-art approach [18], [19]. Another common solution for UAV navigation is the use of landmarks [20]. Hummel et al. [21] present bookshelf scanning with a drone, based on rectangle detection validated with book mock-ups on a poster. In summary, although a wide research in navigation has taken place [22], [23] autonomous indoors UAV navigation is still a challenge.

Recapitulating, ground based robots have been and already are being used in libraries, but automation based on aerial robots has been not so far due to the above-mentioned technological challenges. Our proposal is that an UAV can be used for library inspection leading to several advantages over human manual inspection such as saving time and cost, easy access to all bookshelves, reading several books per image, and alerting to book misplacements in real-time. To the best of our knowledge the use of drones has not been reported as an automated inventory device within libraries. In this paper, we present the UJI aerial librarian robot, an unmanned quadrotor drone -or quadcopter- that leverages computer vision techniques to autonomously self-localise and navigate within a library for automated inventory and book localisation, identifying misplaced books. Our quadcopter robot has been tested in a real library with promising results.

II. Localisation

We used the Parrot Bebop 2 quadcopter [24] (see Figure 1) and a self-tracking approach based on visual markers in order to compute the quadcopter position in the library. To this end, several ArUco markers [25] were placed on the bookshelves in specific poses (see Figure 2) so that the quadcopter pose is estimated according to the number and type of visible markers.

Fig. 1. Parrot Bebop 2 quadcopter

Fig. 2. A bookcase sample equipped with ArUco markers

Note that the precision of the quadcopter pose estimation directly depends on the knowledge about the markers pose, since the camera is rigidly mounted on the drone. It is necessary to introduce a calibration process. With that aim, visual information from the camera is used to calculate mutual poses of the markers as follows:

1) The three markers closest to the origin of the global coordinate system (let call them *initial markers*) are chosen, manually measured and preset their poses in global coordinates in a Cartesian coordinate system that we arbitrarily set up (see Figure 3)
2) Mutual poses of the initial markers are searched and, from that information, their preset global poses are corrected not to contradict camera-measured mutual poses
3) On the basis of corrected poses of the three initial markers, positions of all other present markers are obtained

Mutual marker poses can be obtained because the marker-tracking library (OpenCV [26] in our case) provides the pose in the camera coordinate system for each marker in every frame. So, if two markers, with index numbers A and B, are present in a frame, two three-component position vectors $P_C^{M_A}$ and $P_C^{M_B}$, and two orientation quaternions $Q_C^{M_A}$ and $Q_C^{M_B}$ (in camera coordinates) are obtained. From these four vectors, transformation matrices $T_C^{M_A}$ and $T_C^{M_B}$ are derived according to Equation 1.

$$T_C^M = \begin{bmatrix} R_C^M & P_C^M \\ 0 & 1 \end{bmatrix} \tag{1}$$

where R_C^M is the rotation matrix obtained from the orientation quaternion Q_C^M following Equation 2 as defined in [27].

The next step is to find the camera pose in the marker coordinate system through the inverse matrix:

$$T_C^{M^{-1}} = \begin{bmatrix} R_C^{M^T} & -R_C^{M^T} \cdot P_C^M \\ 0 & 1 \end{bmatrix} \tag{3}$$

Thus, the transformation matrix for conversion from the reference of one marker into the reference frame of the other marker is obtained as follows:

$$T_{M_A}^{M_B} = T_{M_A}^C \cdot T_C^{M_B} \tag{4}$$

Then, $T_{M_A}^{M_B}$ is divided into $P_{M_A}^{M_B}$ and $Q_{M_A}^{M_B}$. With the purpose of getting a higher precision in mutual pose estimation, several images of the same pair of markers are taken from different points of view and the average is considered for the quadcopter localisation.

This process is performed for each appropriate pair of markers when more than two markers are present in the frame. The markers are sorted in ascending order according to their index numbers, that are all different. So, as illustrated in Figure 3, the index number of markers increases as markers get placed further from the origin of the global coordinate system. Bundle adjustment is used for the three initial markers; poses for every of the "further" markers are calculated as an average based on the positions obtained from several "nearer" markers. A weighted average of the pose considering several markers falling into the field of view was used to increase the stability of localisation with respect to the robot map.

III. Navigation

The possibility to control the velocity of the quadcopter (i.e. linear X speed, linear Y speed, linear Z speed, angular yaw speed) is implicitly presented by software-hardware implementation of Parrot Bebop 2. In addition, the quadcopter has difficulties indoors to keep the same position for a long time when no movement commands are given.

Consequently, it is required to continuously adjust the movement commands based on the position feedback. First, a position control module is in charge of ensuring that the quadcopter is moving towards the goal point adequately, or that it does not drift away from the goal point when it is already there. For that, it estimates and sends the Y linear translation ($v_{y,d}$), roll rotation (γ_d), pitch (θ_d) and yaw ($\omega_{\psi,d}$) angles to the quadcopter (see Figure 4). So, proportional control can be applied to control Y coordinate and yaw angle of the quadcopter in accordance with Equation 5.

$$\begin{aligned} v_{y,d}(t) &= K_{P,y} \cdot e_y(t) \\ \omega_{\psi,d} &= K_{P,\psi} \cdot e_\psi(t) \end{aligned} \tag{5}$$

where $K_{P,y}$ and $K_{P,\psi}$ are predefined proportional coefficients, while $e_y(t)$ and $e_\psi(t)$ represent the errors between the actual and desired pose of the quadcopter in terms of the corresponding coordinate. When a similar principle was considered with roll and pitch angles to control X and Z coordinates, unsatisfactory results were obtained. The main reason is that X and Z quadcopter speeds are not proportional to roll and pitch angles, as it can be derived from a simple dynamical model of the quadcopter. Instead, there is a dynamical connection between the values and it is possible to take this into account by adding a differential component to the controller. Hence,

$$R_C^M = \begin{bmatrix} 1-2y_C^{M^2}-2z_C^{M^2} & 2x_C^{M^2}y_C^{M^2}+2w_C^{M^2}z_C^{M^2} & 2x_C^{M^2}z_C^{M^2}-2w_C^{M^2}y_C^{M^2} \\ 2x_C^{M^2}y_C^{M^2}-2w_C^{M^2}z_C^{M^2} & 1-2x_C^{M^2}-2z_C^{M^2} & 2y_C^{M^2}z_C^{M^2}+2w_C^{M^2}x_C^{M^2} \\ 2x_C^{M^2}z_C^{M^2}+2w_C^{M^2}y_C^{M^2} & 2y_C^{M^2}z_C^{M^2}-2w_C^{M^2}x_C^{M^2} & 1-2x_C^{M^2}-2y_C^{M^2} \end{bmatrix} \tag{2}$$

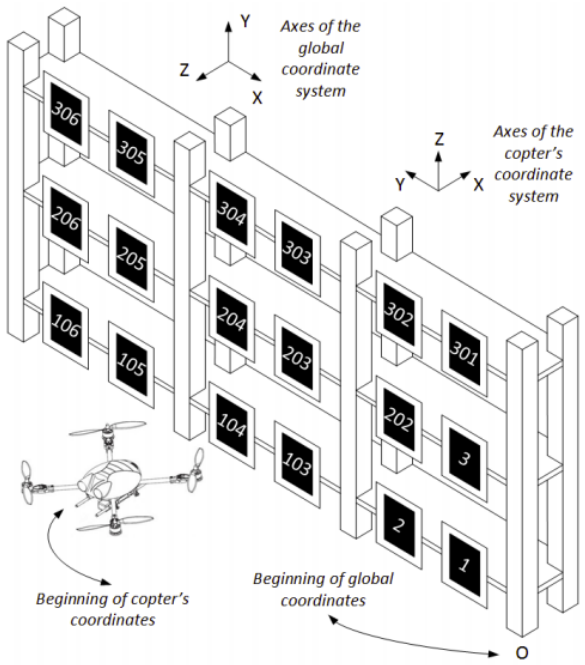

Fig. 3. Pattern for placing the markers on the bookcase based on their index numbers. Note also the orientation of the two coordinate systems

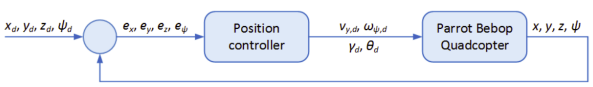

Fig. 4. Flowchart of our implemented position control system where x, y, z, and ψ represent the actual pose of the quadcopter; x_d, y_d, z_d, and ψ_d represent its current desired pose; e_x, e_y, e_z, and e_ψ represent the error between the actual and the desired pose of the quadcopter; $v_{y,d}$, γ_d, θ_d and $\omega_{\psi,d}$ represent the commands sent from the controller to the quadcopter

the desired values for pitch and roll angles can be calculated according to Equation 6, obtaining more accurate results as shown in Figure 5.

$$\begin{aligned} \gamma_d(t) &= K_{P,x} \cdot e_x(t) + K_D \cdot \frac{de_x(t)}{dt} \\ \theta_d(t) &= K_{P,z} \cdot e_z(t) + K_D \cdot \frac{de_z(t)}{dt} \end{aligned} \tag{6}$$

Once the quadcopter is able to reach the desired points in 3D space, the next step is path planning. In this case, the quadcopter should fly along each shelf following a linear path. As illustrated in Figure 6, the followed trajectory departs from the straight line, although all the waypoints are reached successfully. Thus, several experiments were carried out to determine the appropriate number of waypoints so that the

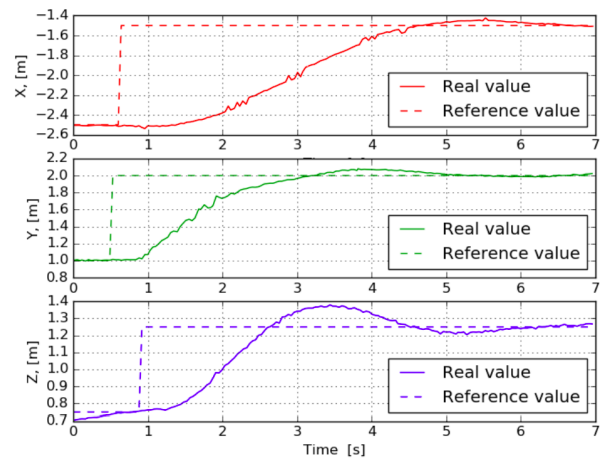

Fig. 5. Transition processed for controlled change of quadcopter position during flight. Graphs for the three axes are independent and represent different moments in flight. For each graph, when transition processing is taking place, reference values for the other axes remain constant.

bookshelves are completely covered. In our case, 8 waypoints for each shelf are required.

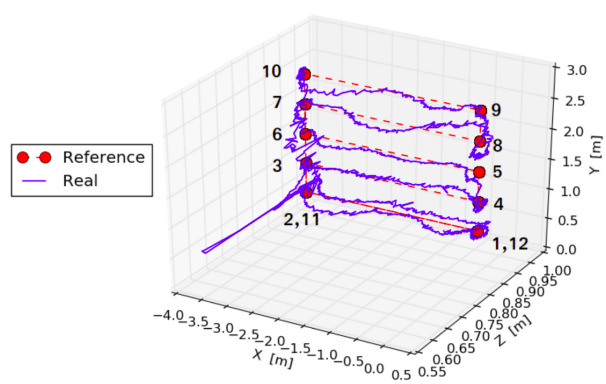

Fig. 6. Experimental trajectory of the Parrot Bebop 2 with 12 waypoints located along the bookcase in the global coordinate system.

IV. Book Recognition

With the aim to perform librarian tasks, the quadcopter must be able to recognise each book within its field of view. For that reason, an analysis of the visual features to properly identify each book is required. However, there are a wide range of visual features that vary from one book to another such as size, thickness, colour and title style. This makes the book recognition a difficult task to achieve.

Deep Learning (DL) techniques could be considered but they need a large set of labeled data. Due to the great amount of books in a library, this process and its corresponding

training stage would be highly time-consuming and unfeasible. Also, RFID systems will not provide information about the exact location of each book.

Instead, our approach takes advantage of the ordinary tags for book cataloguing that are used in most of the libraries worldwide. Basically, a tag is a homogeneously coloured label where an alphanumeric code according to the library catalogue is written in black. These tags are usually placed at the bottom of the book spines, as shown in Figure 8. On this basis, we have designed a novel aerial vision approach for book recognition that is composed of the detection of the book tags, followed by the book code on each tag by means of Optical Character Recognition (OCR).

A. Book tag detection

The first step is to automatically define a region of interest, that is, focus the search in the area where the books are instead of the whole image. For that, the ArUco markers are used. So, all the ArUco markers included in the considered dictionary are searched in each frame so that we obtain a list of all the 2D corner positions corresponding to each detected marker together with its identifier. From these data, a horizontal line joining the top marker corners is defined. This line represents the base of the considered region of interest, which is 200 pixels high, as illustrated in Figure 7. The resulting image is converted to grayscale as follows:

$$gray = 0.299 * R + 0.58 * G + 0.114 * B \tag{7}$$

Next, the image is filtered in order to remove noise. In particular, a 9x9 Gaussian kernel is used. The subsequent stage corresponds to edge detection; for that, a threshold for each region of the image is estimated by means of a combination of binary thresholding with the Otsu thresholding. The use of local thresholds instead of a global one for the whole image provides better segmentation results for images with varying illumination -a common situation in a library. Then, an erosion operation allows the system to extract the vertical lines identifying the book borders. Similarly, an adaptive thresholding followed by a pair of morphological operations allows to identify the horizontal edges of the book tags.

Once all the edges have been properly distinguished, the intersection points are searched. Finally, the book tags are obtained from those intersection points. Indeed, superimposing the lines on the original image confirmed the detection of book tag boundaries.

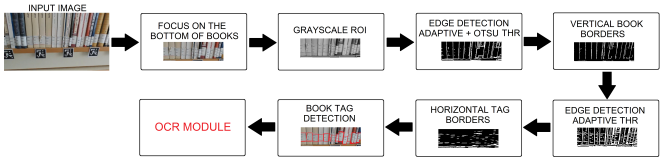

Fig. 7. Flowchart of our vision approach for book tag detection

Some of the obtained experimental results are shown in Figure 8. As illustrated, the designed approach properly detects the book tags in most of the cases, even when they are partially broken or the book is inclined. However, the approach fails when the book tag is not placed at the bottom of the book spine, as in the case of Figure 8c. Another error takes place when two consecutive books are too similar and there is no clear vertical space between them (see Figure 8d). As an overall result of several runs, since often a tag that is incorrectly identified in one frame is successfully detected in the next, overlapping frame, the actual success rate is that 85 % of the book tags in the bookshelves are correctly detected.

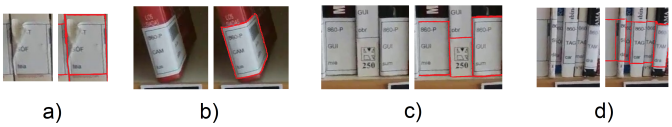

Fig. 8. Some results of the book tag detection illustrating successful results for partially broken (a) and slanted tags (b), and problematic cases such as too high tags (c) and unclear vertical separation (d)

B. Optical Character Recognition (OCR)

In this work, Pytesseract, a wrapper for Tesseract-OCR Engine, was used. So, each tag detected by the book tag detection module is fed into Pytesseract for its processing. Then, the generated output is checked in terms of its consistency, since only partial text could have been provided due to broken book tags or the book thinness. In addition, skewed or rotated text may also make the OCR fail. With all this, 75 % of the books in the bookshelves were detected correctly.

V. Experimental Results

The UJI aerial librarian robot was tested in the library of the high school I.E.S. Els ports. In the experiments, the robot navigated around wooden bookcases with a height of 2.4 metres, a width of 1.7 metres and a depth of 40 centimetres (see Figure 2). Each bookcase is composed of 6 bookshelves. Each bookshelf was equipped with several visual 7x7 ArUco markers necessary for localisation and navigation. In particular, they were attached to the front side of each shelf, with a horizontal distance between the markers of some 7 centimetres. Such placement resulted in 8 markers per shelf and a total of 48 markers per bookcase.

After the experimental set-up, the gathering of book information (i.e. global localisation and book recognition) was run for several times. Several conditions were considered such as misplaced books, occlusions, different positions in depth and so on. The capture of several images from the same position usually improves the results of book recognition. As overall result, the average rate of book recognition and localisation was 65 %, (see Figure 9).

Fig. 9. Some results of the book recognition through a bookcase

VI. Conclusions

In this paper, we present the UJI aerial librarian robot, an autonomous quadcopter that is able to globally localise books in an ordinary library by leveraging an aerial vision approach for localisation, navigation and book recognition. It is the first such a robot ever, to the best of our knowledge.

Our experimental study was conducted in a real scenario. After several tests with varying conditions, the accuracy rate was 65 %. If we do not take into account the books with unreadable tags (due to damaged tags or too thin book spines) -since not even humans can read them- this rate increases to 72 %. We believe these are very promising results, for a first ever proof-of-concept system, that opens the way to further research along this path. In addition, images were processed on line, allowing for real-time applications.

In addition, this work improves on state-of-the-art approaches to other applications of UAVs. First, our ad-hoc control strategy for autonomous flight is aimed at overcoming two main problems: an automated collision-free navigation in an indoor environment avoiding interferences such as Wi-Fi, or telephony; and an accurate book inspection through the different bookcases composing the library, while a continuous motion takes place. Our approach combines different computer vision algorithms to accurately detect and extract the book tags for their recognition. In this case, our contributions pertain to the detection and recognition of books with different visual features (e.g. colour and size), as well as the suppression of requirements such as specific distance between camera and book, book orientation, or camera orientation with respect to the bookshelf.

For future work, higher quality visual sensors are required for better performance. Even so, for a 65 % success rate, we estimate that the inventory time would be reduced to a 42 % of the fully manual time cost, considering that the required time is around one order of magnitude smaller for the UJI Aerial Librarian Robot.

Acknowledgment

Support for our research is provided in part by Ministerio de Economía y Competitividad (DPI2015-69041-R), by Universitat Jaume I (UJI-B2018-74), and by Generalitat Valenciana (PROMETEO/2020/034, GV/2020/051).

References

[1] E. Martinez-Martin, G. Recatala, and A. P. del Pobil, "Transforming library operation with robotics," in *Information Technology Satellite Meeting "Robots in libraries: challenge or opportunity?" (IFLA WLIC)*, Berlin, Germany, aug 2019.

[2] M. Morenza-Cinos, V. Casamayor-Pujol, J. Soler-Busquets, J. L. Sanz, R. Guzmán, and R. Pous, "Development of an RFID inventory robot (AdvanRobot)," in *Studies in Computational Intelligence*. Springer International Publishing, 2017, pp. 387–417.

[3] R. Li, Z. Huang, E. Kurniawan, and C. K. Ho, "AuRoSS: An autonomous robotic shelf scanning system," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, sep 2015.

[4] I. Ehrenberg, C. Floerkemeier, and S. Sarma, "Inventory management with an RFID-equipped mobile robot," in *2007 IEEE International Conference on Automation Science and Engineering*. IEEE, sep 2007.

[5] N.-H. Quoc and W.-H. Choi, "A framework for recognition books on bookshelves," in *Emerging Intelligent Computing Technology and Applications*. Springer Berlin Heidelberg, 2009, pp. 386–395.

[6] Z. Hu, J. Tang, and L. Lei, "A hybrid algorithm for the segmentation of books in libraries," in *Mobile Multimedia/Image Processing, Security, and Applications 2016*, S. S. Agaian and S. A. Jassim, Eds. SPIE, may 2016.

[7] N. Tabassum, S. Chowdhury, M. K. Hossen, and S. U. Mondal, "An approach to recognize book title from multi-cell bookshelf images," in *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*. IEEE, 2017.

[8] M. I. Jubair and P. Banik, "A technique to detect books from library bookshelf image," in *2013 IEEE 9th International Conference on Computational Cybernetics (ICCC)*. IEEE, jul 2013.

[9] P. Prashanth, K. S. Vivek, D. R. Reddy, and K. Aruna, "Book detection using deep learning," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE, mar 2019.

[10] B. Zhu, X. Wu, L. Yang, Y. Shen, and L. Wu, "Automatic detection of books based on faster r-CNN," in *2016 Third International Conference on Digital Information Processing, Data Mining, and Wireless Communications (DIPDMWC)*. IEEE, jul 2016.

[11] X. Yang, D. He, W. Huang, Z. Zhou, A. Ororbia, D. Kifer, and C. L. Giles, "Smart library: Identifying books in a library using richly supervised deep scene text reading," *CoRR*, vol. abs/1611.07385, 2016.

[12] M. Prats, E. Martinez, P. J. Sanz, and A. P. del Pobil, "The uji librarian robot," *Journal of Intelligent Service Robotics*, vol. 1, pp. 321–335, 2008.

[13] D. Unnikrishnan, C. Aswani, A. K. Jayaprakash, and S. Ganesh, "Library assistant robot. robots in library management system," *International Journal of Engineering Research & Technology (IJERT)*, vol. 6, jan 2017.

[14] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[15] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, pp. 460–466, 2015.

[16] E. H. C. Harik, F. Guerin, F. Guinand, J.-F. Brethe, and H. Pelvillain, "Towards an autonomous warehouse inventory scheme," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, Dec 2016.

[17] C. Kanellakis and G. Nikolakopoulos, "Survey on computer vision for UAVs: Current developments and trends," *Journal of Intelligent & Robotic Systems*, vol. 87, no. 1, pp. 141–168, jan 2017.

[18] T. Han, J. S. Almeida, S. P. P. da Silva, P. H. Filho, A. W. de Oliveira Rodrigues, V. H. C. de Albuquerque, and P. P. R. Filho, "An effective approach to unmanned aerial vehicle navigation using visual topological map in outdoor and indoor environments," *Computer Communications*, vol. 150, pp. 696–702, jan 2020.

[19] W. Kwon, J. H. Park, M. Lee, J. Her, S.-H. Kim, and J.-W. Seo, "Robust autonomous navigation of unmanned aerial vehicles (uavs) for warehouses' inventory application," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 243–249, jan 2020.

[20] R. R. Ibarra, M. V. Márquez, G. Martínez, and V. Hernández, "Computer vision navigation system for an indoors unmanned aerial vehicle," in *Communications in Computer and Information Science*. Springer International Publishing, 2020, pp. 30–47.

[21] K. A. Hummel, M. Pollak, and J. Krahofer, "A distributed architecture for human-drone teaming: Timing challenges and interaction opportunities," *Sensors*, vol. 19, p. 1379, march 2019.

[22] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, jan 2020.

[23] T. Cabreira, L. Brisolara, and P. R. F. Jr., "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3, no. 1, p. 4, jan 2019.

[24] Parrot Drone SAS, "Parrot bebop 2 quadcopter," https://www.parrot.com/en, 2015, accessed on: oct 2020.

[25] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[26] OpenCV team, "Opencv," https://opencv.org/, accessed on: oct 2020.

[27] J. M. P. van Waveren, "From quaternion to matrix and back," Id Software, Inc., Tech. Rep., 2005.