Towards An Autonomous Warehouse Inventory Scheme

Frédéric Guinand, Jean-François Brethé, El Houssein Chouaib Harik, François Guerin, Hervé Pelvillain

Towards An Autonomous Warehouse Inventory Scheme

El Houssein Chouaib Harik[1], *Student Member, IEEE*, François Guérin[2], Frédéric Guinand[3],
Jean-François Brethé[2], *Member, IEEE*, Hervé Pelvillain[4]

Abstract— *We present in this paper a novel warehouse inventory scheme. The main purpose of this work is to make the inventory process completly autonomous. To this end, an Unmanned Ground Vehicle (UGV) and an Unmanned Aerial Vehicle (UAV) works together. The UGV is used as the carrying platform, and considered as a ground reference for indoor flight of the UAV. The UAV is used as the mobile scanner. The UGV navigates among rows of racks carrying the UAV. At each rack to be scanned, the UGV stops at the bottom, the UAV takes off and flies vertically scanning goods in that rack. Once at the top, the UGV moves to the next rack, and since the UAV takes the UGV as the ground reference, it will follow it, this results in placing the UAV at the top of the second rack, and scanning from top to bottom starts. the process is repeated until the row of racks is fully scanned, and the UAV lands on the UGV, and recharge its batteries while the UGV moves to the next row of racks. We present in this paper the proposed architecture, as well as the first experimental results of the proposed scheme*

I. INTRODUCTION

Logistics industry is very developed in the region of Le Havre, France, but this domain is also a development axis of le Havre University, that managed the construction of a dedicated infrastructure for that purpose: the PIL. This activity center gathers the related technological activities to the field. One of the main activities of the PIL center is developing innovative solutions for warehouses, from autonomous packaging services, to complete autonomous warehouses.

One of the cumbersome tasks in warehouses is the inventory. It needs the mobility of a lot of resources especially when the warehouse belongs to big supply chains, where racks can easily reach the height of 10 meters, which may generate delays in information update due to time consumption of the task, as well as probable injuries of the workers performing the inventory when located at great heights. Adding to that the high costs of the operation because of both, the inventory workers, and the blocked activities of the warehouse during this inventory. Many efforts has been made for automating the inventory process. We will focus in this paper on the solution that uses UAVs as the scanning agents.

Their mobility and ability to carry small optical scanners allowed the UAV to be a great candidate to replace humans

[1]EHC. Harik is with LITIS, University of Le Havre, France. E-mail: el-houssein-chouaib.harik@univ-lehavre.fr.

[2]F. Guérin and J-F. Brethé are with GREAH, University of Le Havre, France. E-mail:(francois.guerin,jean-francois.brethe)@univ-lehavre.fr.

[3]F. Guinand is with LITIS, University of Le Havre, France, and UKSW, University Cardinal Stefan Wyszynski in Warsaw. E-mail: frederic.guinand@univ-lehavre.fr.

[4]H. Pelvillain is with IUT GEII, University of Le Havre, France. E-mail: herve.pelvillain@univ-lehavre.fr

in warehouse inventory. On long term, they represent a great economic solution for the task, as well as the diminution of time consumption and injuries risks of the task. Nevertheless, this technology is still at its early stage, and even though some companies advertise solutions for that task [1], [2], [3], the platforms are whether research projects, or use manually piloted UAV.

We propose in this paper a novel scheme towards the automation of the warehouse inventory task. We use an UAV to replace the human operator for the scanning purposes. A warehouse is a GPS denied environment, thus inside flight of the UAV is the first challenge to be solved. One can think of using local geo-localization methods to locate the UAV inside the warehouse, but obstacle avoidance and scanning the racks still represent another challenge in order to have a fully autonomous platform. Thus the main objective for a fully autonomous platform for warehouse inventory tasks, is to have an UAV capable of inside autonomous flight, obstacle avoidance, rack scanning, and autonomous landing and take-off. The collected data should be visualized and stored in real time.

We presented in a [4] a work in which using a UGV as a ground reference, the UAV is able of autonomous flight with good tracking results. Build on this work, we can imagine using the UGV as a ground reference for the indoor flight. This choice is mainly justified by the high cost, and cumbersome implementation of the exteroceptive localization solutions in GPS denied environments. The UGV navigates among rows of racks carrying the UAV, at each rack to be scanned, it stops at its bottom, the UAV takes of, flies vertically scanning the goods in that racks. Once at the top, the UGV moves to the next rack, since the UAV takes the UGV as a ground reference, it will follow it, which will locate the UAV at the top of the second rack. The UAV now scans the rack inversely, from top to bottom. The process is repeated until the row of racks is fully scanned. At the end of that row, the UAV lands on the UGV to save, and eventually recharge its batteries, while the UGV moves to another row.

The rest of the paper is organized as follows: In section II we give an overview of the global architecture of the proposed scheme. We present in section III the vision modules of the systems, mainly for the the autonomous tracking of the UGV and the the rack IDentification (ID) detection. We detail in section IV the control law that enables the UAV to follow autonomously the UGV. Some experimental results will be presented in V. We conclude in section VI the present work and give some future directions and perspectives.

II. GLOBAL ARCHITECTURE

The UGV is used as ground reference for the UAV for indoor flight. For this purpose, a down facing camera is mounted beneath the UAV for vision based target tracking. The UGV is detected from the down facing camera module using an Augmented Reality (AR) marker placed on the planar surface of the UGV. The position of the AR marker in the image is fed to the UAV controller in order to regulate its pitch and roll angles to stay upon the UGV when flying. A front facing Barcode scanner mounted on the UAV is used for goods scanning, and the information is sent via a wireless communication in real time to the ground station for the inventory purposes.

The UGV is equipped with lidars for navigating among row of racks. It uses the output of these sensors in order to maintain a fixed distance when navigating among the rows. At the bottom of each row, a coded marker that indicates the ID of the corresponding rack is placed. The UGV is also equipped with front facing camera to enable it to recognize these marker and stops at each one for them.

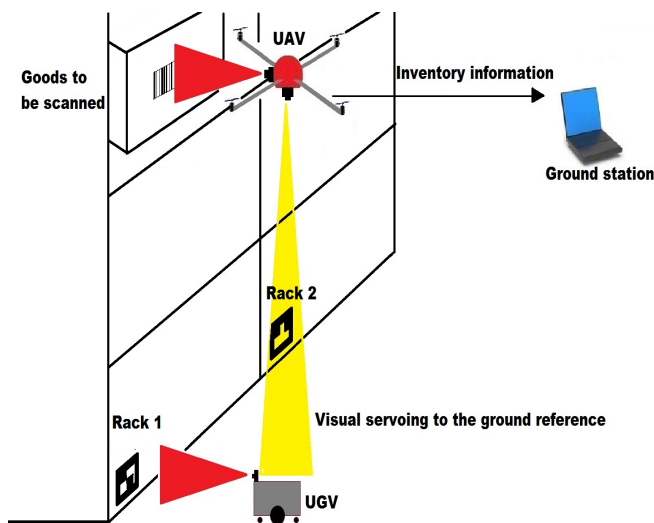Figure 1 illustrates the global architecture of the proposed scheme.

Fig. 1.   Autonomous warehouse inventory global architecture

The UAV is initially landed on the planar surface of the UGV. The latest navigates within rows and stops at the first rack to be inspected. Once the UGV arrives at *Rack 1* (figure 1), it stops, the UAV takes off, and start flying vertically up scanning the Barcodes on the goods placed at each level of the rack. The height of racks are predefined, once the UAV reaches the top of *Rack 1*, an information is sent to the UGV so it starts moving to the *Rack 2*. Once the UGV detects the AR code of *Rack 2*, it stops. Meanwhile, the UAV using the down facing camera changes its roll and pitch angles to stay on the top of the UGV, which means that now the UAV is placed on the top of the second rack. The UAV starts flying vertically down scanning the goods until it reaches the first element on *Rack 2*. The UAV sends the signal to the UGV

to move to the *Rack 3*, and the process is repeated until a row is finished.

Once a full row is scanned, the UAV lands on the UGV to recharge its batteries while the UGV moves to the next row to be scanned.

III. VISION MODULES

To track autonomously the UGV, the UAV uses its down facing camera to detected the AR marker placed on its planar surface. The usage of the AR markers is inspired by an Augmented Reality (AR) application initiated in [5]. The authors presented a vision based algorithm to estimate the pose of a known marker (the AR marker) in the space, and overlay different graphical shapes on it. The proposed method is publicly available as a library known with the name of *ARToolKit*.

This AR application inspired many members of mobile robotic research community. Mainly to give a better alternative to color tracking for pose estimation. A printed AR marker placed upon a mobile robot (for instance UGV). The library for the extraction of the marker position in the image outperformed color technique for pose estimation, especially for the immunity to lightning conditions (as long as there is enough light to distinguish the marker). As an example, this method was included in [6] to spot several UGVs for a vision feedback control loop. A similar effort in using AR for target detection can be found in [7]. The presented method allows the tracking of several features at a time with a good precision using low cost cameras, and uses the same kind of visual marker as in ARToolKit. The proposed method has been made publicly available as a library called ArUco [8]. Another AR marker tracking can be found in [9]. The authors used a down facing camera mounted on an aerial vehicle (quadrotor) in order to hover it on a ground feature (the AR marker). They fused the information gathered from the vision combined with internal sensors (Inertial Measurement Unit (IMU)) in order to generate the necessary movements to stay on the top of the ground feature.
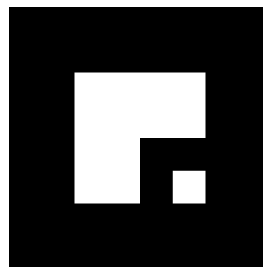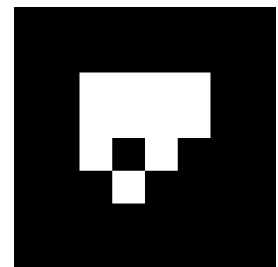

Fig. 2.   Marker 1

Fig. 3.   Marker 2

We use in our work the library presented in [5]. Figures 2 and 3 represent two examples of AR markers [10] that can be printed and used later for the UGV's pose estimation. Each marker comes with a different Bitmap IDentifier, which allows this technique to be used as well for racks identification by the UGV, where we assign a different marker for each rack.

Each image received from the camera mounted on the UAV have to be processed to check the presence of the predefined AR marker. The AR marker is placed at the upper visible surface of the UGV. The surface is plan, without deformation, and the UGV is supposed to navigate on a planar surface, so no distortion of the physical marker is present. The center of the AR marker is supposed to be the center of the UGV.

The library presented in [5] is used to extract the coordinates of the AR Marker. The library works as follows: the received image is thresholded to extract contours. After extracting the contours inside the image, another process is done: if a contour matches a square shape (fitted in four line segments), the sub-image inside the square will be compared to the pattern of AR marker. If a match is found with the stored AR marker, the corners of the square are saved, and used for any further process.

We consider X_D and Y_D the axes of the image taken from the drone (Figure 4). In order to locate the UGV, we will use the presented tracking process to extract the UGV's position (R_c). The position (in pixels) of R_c along X_D and Y_D axes is: Y_{Rc} and X_{Rc}. Y_0 and X_0 are the center (0, 0) of the image (we assume that the center of the image is the center of the drone).
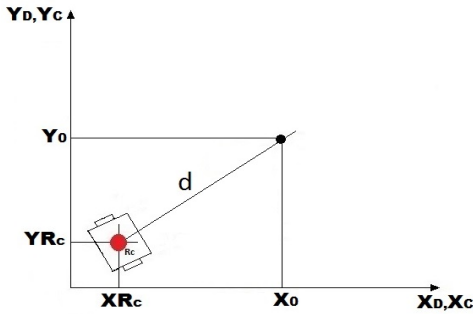
Fig. 4.   The UGV in the image plan

The distance d that separates the UGV and the center of the UAV can be written as follows:

$$d = \sqrt{(X_0 - X_{Rc})^2 + (Y_0 - Y_{Rc})^2} \qquad (1)$$

Where X_0, Y_0, X_{Rc}, Y_{Rc} represent the coordinates of the marker R_c and the center of the aerial vehicle in its (or camera) frame previously supposed to be the same. The pose estimation of the UGV from the visual input is directly defined in the camera space, thus no need for transforming the coordinates in a world frame.

Algorithm 1 illustrates the sequences of the necessary steps in order to extract UGV's center (R_c) coordinates in the image.

After receiving the video flow on the ground station, the algorithm is applied in order to extract the features of the AR marker (figure 2).
corner[i].x and corner[i].y for i = 1..4, are the 2D coordinates of the extracted ar marker corners. the methods

---

Algorithm 1: Augmented reality marker tracking

Input: Video flow, AR marker (the tracked coded marker)
Output: X_{R_c}, Y_{R_c} (2D pose of the tracked marker)
begin
    Define AR marker (marker 1) ;
    image = videoFlow.getNextImage() ;
    while *image != null* **do**
        if *hasMarker(image,marker 1)* **then**
            /*extract marker 1 corners*/;
            corners[]=image.ExtractMarkerCorners();
            /*determination of the center of the marker 1*/;
            dx1 = x.corner[2]-x.corner[0];
            dy1 = y.corner[2]-y.corner[0];
            m1=dy1/dx1;
            c1=y.corner[0]-m1*x.corner[0];
            dx2 = x.corner[3]-x.corner[1];
            dy2 = y.corner[3]-y.corner[1];
            m2=dy2/dx2;
            c2=y.corner[1]-m2*x.corner[1];
            X_{R_c} = (c2 - c1) / (m1 - m2);
            Y_{R_c} = m1 * X_{R_c} + c1;
            draw a red circle on (X_{R_c}, Y_{R_c});
        image = videoFlow.getNextImage();

---

getNextImage() are provided by the UAV video extraction library, while the method *hasMarker()* is provided by the AR library presented in [5], and works as we previously explained.

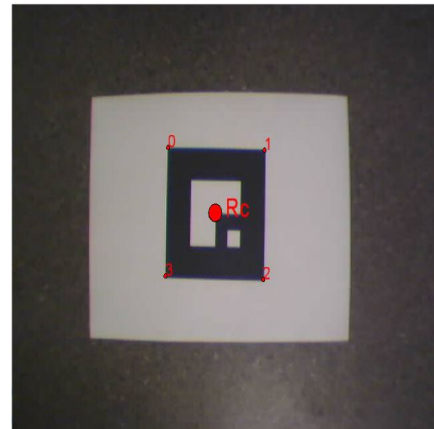Figure 5 shows the results of the AR marker tracking algorithm.

Fig. 5.   Augmented reality marker detection

For scanning the goods on the racks, the UAV is equipped with a commercially available, light weight Barcode scanner. the gathered information is sent in real time through a wireless communication to the ground station, where it will be processed for the inventory tasks.

Algorithm 1 is used on the UGV to detect the racks IDs. These information are fed into the UGV embedded computer to know its position in the warehouse according to the rack ID in front of it. To navigate among racks, the UGV uses its lidars to keep the same distance to a given row. Since the UGV is the ground reference, this distance is estimated to avoid any collisions of the UAV with the racks.

IV. AUTONOMOUS FLIGHT CONTROLLER

When moving from a rack to the next one, the UAV follows autonomously the UGV. For this purpose we developed control laws to change the pitch and roll angle of the UAV (having a yaw angle fixed). To stay on the top of the UGV, the input of the UAVs' controller are the location of the marker placed on the top of the UGV (X_{Rc}, Y_{Rc}). The vertical flights is performed at a fixed speed to allow the scanning of the goods from bottom to top, or top to bottom using the altitude sensors of the UAV, thus the total height of the racks inside the warehouse should be known.

A. Fuzzy Logic Controller with Double Smoothing Algorithm

We have developed in previous works [4] and [11] model based non linear controllers for vision based target tracking of UAV. In order to have an accurate autonomous tracking of the UGV, the most accurate physical model of the UAV should be used. Unfortunately, the parameters are not provided by UAV manufacturers, this is the reason why we used rough estimated parameters, resulting in rough tracking results. For controlling such systems we think that a Fuzzy Logic Controller (FLC) is very suitable. Indeed, as said by Prof Zadeh who introduced this concept in 1965, fuzzy logic is an "attempt to mimic human control logic" and this matches our goal of emulating the behavior of a professional pilot controlling manually a UAV over a moving UGV using visual feedback.

Fuzzy Logic Controllers (FLC) have been widely used for vision based tracking. The tracked target location is extracted by estimating the pose of a unique visual feature that belongs to the target. A simple way to do so is to track a color. A red balloon is used in [12] to simulate a flying object. The distance and the orientation to the target are estimated using the size and the position of the balloon in the image plan. Artificial markers (AR markers for example) can also be used to estimate the pose of the tracked target. They are usually placed on a planar surface on the target, and the inputs to the FLC are the X and Y errors between the center of the image and the center of the marker. Vision based FLC is used in other works for navigation [13], [14], and autonomous landing [15], [16].

A first step in creating a FLC to control the UAV consists in creating fuzzy sets. A fuzzy set is the entity that represents an input or an output of a system. In our case, the input is the pose estimation of the UGV in the image plan, and the output is the desired roll and pitch angles in order to follow autonomously the UGV. Taking the example of an input, the fuzziness of the set is illustrated in the degree of belonging to the different states of this set. In the UAV image plan,

the UGV position can be at a far negative (N), at the center (Z), or at a far positive (P) distance from the center of the image. These are called membership functions, where we define the degree of belonging of the crisp values of the UGV pose to these states (N, Z, P). Once we have our input membership functions, the same process is applied to extract the output, and the bridge between the two values (input and output) is what is called Defuzzification. We will see in next section more details on the different membership functions of the FLC and their defuzzification in order to get the desired attitude of the UAV for the autonomous target tracking.

As the UGV is supposed to move on a planar surface, the two outputs that needs to be fuzzified in order to follow autonomously the UGV are roll and pitch angles, which are respectively the angles necessary to change in order to move along X and Y axes of the UAV. The input to the FLC is then location of the UGV in the image plan, which is in our work the AR marker X and Y position of its center obtained as explained in previously.

Once the UGVs location extracted, the pose errors to the center of the image are estimated. In real application, the measurement of the markers center is noisy, which may lead to unpredictable behavior of the UAV during the autonomous flight tracking.

1) Double Exponential Smoothing algorithm: For smoothening the noisy measurements provided by the vision system, a Kalman filter could be implemented, we have however chosen the Double Exponential Smoothing (DES) algorithm [17]. indeed, compared to a Kalman filter, the DES algorithm runs approximately 135 times faster with equivalent prediction performances and simpler implementations [17] .

If T_e denotes the sampling period, n the discrete-time index, and S_{m_n} the smoothed values of X_{RC} and Y_{RC} the projection of center of the AR marker, then, following the DES algorithm [18], S_{m_n} is computed according to the following equation :

$$S_{m_n} = \gamma_m.m_n + (1 - \gamma_m).(S_{m_{n-1}} + b_{m_{n-1}}) \quad (2)$$

The trend values:

$$b_{m_n} = \lambda_m.(S_{m_n} - S_{m_{n-1}}) + (1 - \lambda_m).b_{m_{n-1}} \quad (3)$$

Where:
m_n is the value of (d,α,X_{Rc}, Y_{Rc}) at n^{th} sample instant.
S_{m_n} is the smoothed values of (d,α,X_{Rc}, Y_{Rc}).
b_{m_n} is the trend value of (d,α,X_{Rc}, Y_{Rc}).

Equation (2) smooths the value of the sequence of measurements by taking into account the trend, whilst (3) smooths and updates the trend.

The initial values given to S_{m_n}, b_{m_n} are:

$$S_{m_1} = m_1 \quad b_{m_1} = m_2 - m_1 \quad (4)$$

Usually, γ ($0 \leq \gamma \leq 1$) is called the data smoothing factor and λ ($0 \leq \lambda \leq 1$) is called the trend smoothing

factor. A compromise has to be found for the values of γ and λ. High values make the DES algorithm follow the trend more accurately whilst small values make it generate smoother results. The smoothed values are used instead of the direct noisy measurements in the proposed controllers. DES algorithm has been used successfully in a previous work [18] for vision based target tracking.

Double Exponential Smoothing (DES) algorithm is used to filter the measurements, and to give a predicted value of the pose estimation. The output of the DES algorithm are fed to the FLC in order to generate the necessary output to track the target. The generated roll and pitch commands are then sent through the chosen communication protocol to the UAV.

We explain the fuzzification of the different inputs and outputs of the controller, and the method used for defuzzification.

2) Inputs and outputs fuzzification: To enable the autonomous tracking of the UGV, the acquired video flow from the vertical camera of the UAV is processed. The results, as explained in algorithm III, are the X and Y coordinates of the marker in the image plan. Since we suppose that the optical center of the camera and the center of the UAV are the same, the extracted coordinates are the relative position of the UGV to the UAV in a planar surface. The errors to be considered then are the ones between the center of the UAV (X_0, Y_0), and the UGV coordinates (X_{Rc}, Y_{Rc}):

$$e_{X_D} = X_{Rc} - X_0 \qquad e_{Y_D} = Y_{Rc} - Y_0 \qquad (5)$$

The control objective can be expressed as follows:

$$\lim_{t \to \infty} e_{X_D}(t) = 0 \qquad \lim_{t \to \infty} e_{Y_D}(t) = 0 \qquad (6)$$

To fulfill this objective, the UAV has to move along its X and Y axes, the outputs of the FLC are in this case the roll ϕ_d and pitch θ_d. Two FLCs were designed to achieve the control objective (6). A representation of the proposed autonomous tracking controller is shown in figure 6.
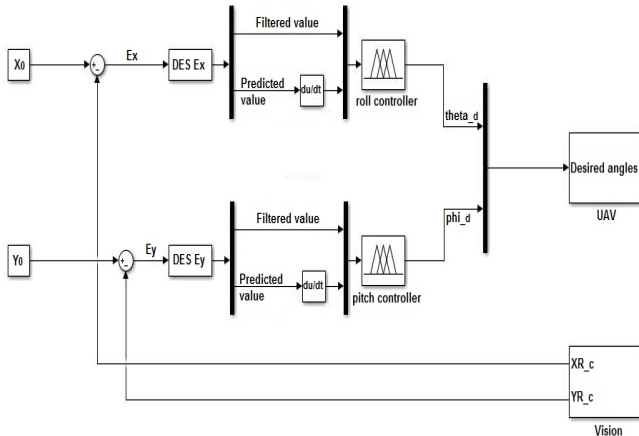
Fig. 6. The structure of the FLC autonomous tracking controller

The UAV is supposed to have a symmetrical construction, pitch and roll controllers are then identical. Thus for brevity, we will present the membership functions of the inputs and output for one of them, the other one is considered to be exactly the same.

3) Membership functions: Roll controller has two inputs, the filtered error and the derivative of its predictive value issued from the DES algorithm described previously.

When maneuvering manually the UAV, the human pilot estimates the position of the UAV according to the one of the UGV when the latest appears at the edge of the screen, thus a big thrust is applied to move the UAV in order to place the UGV in the center of the image. The UAV pilot consider the UGV to be at the center of the image with a certain radius, and apply little thrust to maneuver the UAV. From this estimation of the UAV pilot behavior, the simplest way to represent this is triangular membership functions, for both the inputs and the output of the controllers. These membership gave good performance to achieve the desired objective of the controller (6).

Figures 7, 8, 9, show respectively the membership functions of the filtered (f) value $e_{X_{D_f}}$, the derivative (D) of the predicted (p) value $e_{X_{D_p}}$, and the desired roll angle θ_d.

The fuzzy sets of the inputs $e_{X_{D_f}}$ and $e_{X_{D_p}}$ are:

1) N: Negative
2) Z: Zero
3) P: Positive

The fuzzy sets of the output θ_d are:

1) NB: Negative Big
2) N: Negative
3) Z: Zero
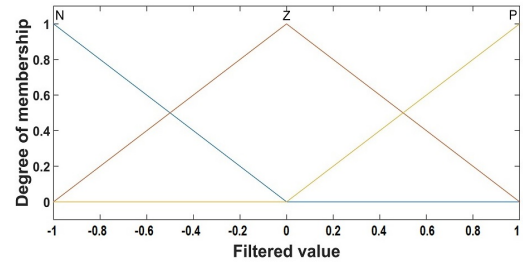4) P: Positive
5) PB: Positive Big
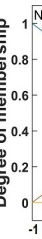
Fig. 7. Membership function of the filtered error
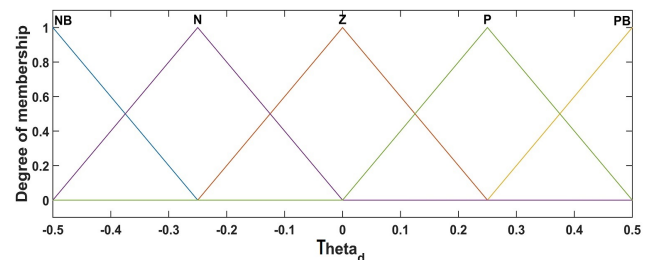
Fig. 8. M

Fig. 9. Membership function of the desired roll angle

4) Defuzzification: We have used Mamdani min-max for the Fuzzy Inference System (FIS). The defuzzification method is centroid based. The FIS has nine rules. Figure 10 shows the FIS inputs and output surface obtained from the Matlab Simulink Fuzzy Logic Toolbox. More details on the impact of the form and number of fuzzy sets have been discussed in [19], [20].
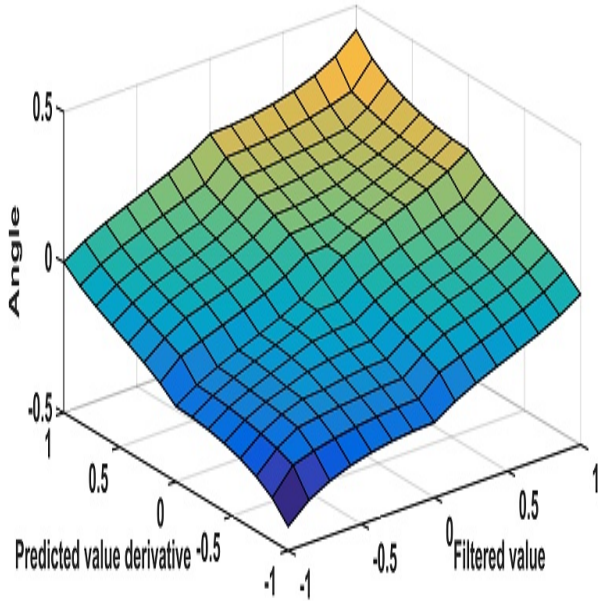

Fig. 10. Fuzzy membership surface for the inputs and the outputs of the controller

Inference of inputs results from the use of "min" operator (for conjunction and implication) and "max" operator (for disjunction and aggregation). Fuzzy rules are summed up in Table I.

e_{XD_f} \ e_{XD_p}	N	Z	P
N	NB	N	Z
Z	N	Z	P
P	Z	P	PB

TABLE I

FLC RULES

Next section is devoted to the presentation of the experimental results of the autonomous flight as well as the preliminary results towards the autonomous warehouse inventory.

V. Experimental results

he following parameters have been used in the experiments:

$maxtilt_r = 0.15\%$, $maxtilt_p = 0.1\%$.

Sampling period: $T_e = 0.01$s.

DES algorithm: $\gamma_m = 0.8$, $\lambda_m = 0.1$, $k_f = 1$.

Please note that $maxtilt_r$ and $maxtilt_p$ are represented in a percentage value of the maxim tilt of the AR Drone given

by the manufacturer, which is 30ř. Figure 11 illustrates the FLC membership functions overview on the GUI developed with Processing. We can see that the tracked marker has been identified in the image, and the estimated errors between its center and the center of the image is fed to the FLC. The input memberships are drawn on the bottom of the GUI (to the left are the inputs of the roll controller, and to the right are the inputs of the pitch controller).
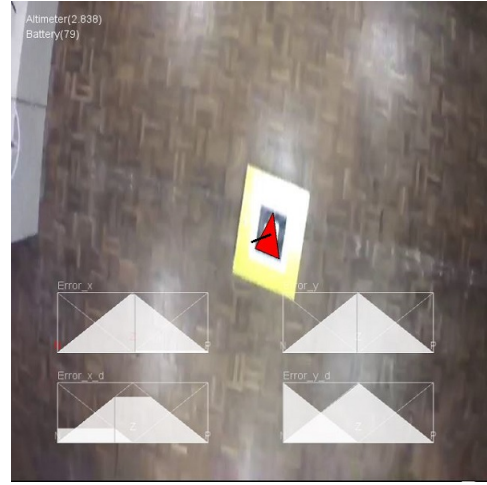

Fig. 11.   The FLC membership functions overview on the GUI

The tracking errors using the FLC with the DES algorithm gave a mean value of around 65 pixels.
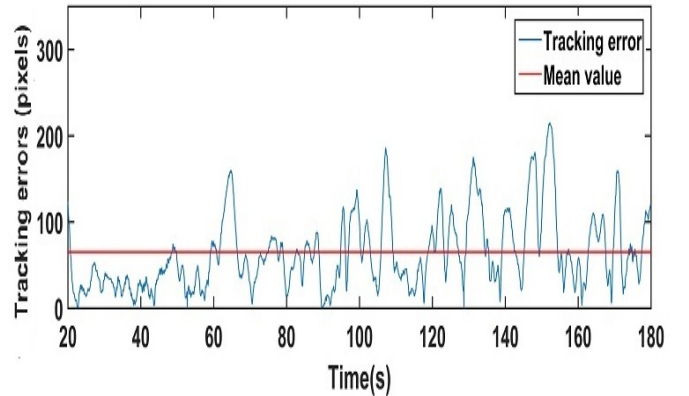

Fig. 12.   Distance error using the FLC with DES algorithm

The usage of the FLC with DES algorithm improved the autonomous vision based controller compared to our previous work in [4] with 47%, which states clearly the efficiency of the proposed controller for the autonomous tracking.

Figure 13 shows the first efforts in this direction. In this experiment, the Summit XL UGV [21] was used as the ground reference. The UGV uses its lidars to navigates at a fixed distance to the wall (emulated rack), and the UAV using its down facing camera will track it autonomously. A front Barcode scanner, or another camera may be used

for scanning the goods on the rack. The UAV will be replaced with a more capable one to perform onboard image processing, which will result in good tracking performance.

Fig. 13. Warehouse inventory using UAV and UGV

VI. Conclusion and Perspectives

We presented in this paper a novel architecture towards an autonomous warehouse inventory scheme. An UAV is used to replace the manual inventory of the available stock. The advantage of using such system is its speediness and efficiency compared to the classical, human-based methods. Since the warehouse inventory is performed in an indoor, GPS denied environment, the UGV is used as a ground reference for the UAV to overcome the expensive, and less practical solutions for indoor localization. The UGV carries the UAV to the desired scanning rack, then the UAV flies vertically to scan Barcodes using the onboard frontal scanner, and send continuously the collected information to the ground station to be processed for the warehouse inventory.

We presented in this work the proposed architecture. We developed autonomous flight controller and shown the efficiency of the results compared to a previous work. Indeed, the warehouse inventory requires more work. For now the UGV navigation is based on the reflection of a flat wall for the lidar sensors, which is not the case with racks, where holes and empty spaces are present in racks, thus further development needs to be done in this direction. The UAV may also need to be equipped with sensors to avoid collision with the racks during the scanning process. Further experiments needs to be performed in a real environment in order to raise other challenges for that application.

VII. Acknowledgments

References

[1] Eyesee: the drone allowing to automate inventory in warehouses, 2016. URL: http://www.hardis-group.com.
[2] Airborne Data Collection, 2016. URL: http://dronescan.co/.
[3] The flying inventory assistant, 2016. URL: http://www.fraunhofer.de.
[4] Harik El Houssein Chouaib, Guérin François, Guinand Frédéric, Brethé Jean-François, and Pelvillain Hervé. A decentralized interactive architecture for aerial and ground mobile robots cooperation. In *IEEE International Conference on Control, Automation and Robotics (ICCAR)*, pages 37–43, Singapore. 2015.
[5] Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 85–94. IEEE, 1999.
[6] Mark Fiala. Vision guided control of multiple robots. In *Computer and Robot Vision, 2004. Proceedings. First Canadian Conference on*, pages 241–246. IEEE, 2004.
[7] Tomáš Krajník, Matías Nitsche, Jan Faigl, Petr Vaněk, Martin Saska, Libor Přeučil, Tom Duckett, and Marta Mejail. A practical multirobot localization system. *Journal of Intelligent & Robotic Systems*, 76(3-4):539–562, 2014.
[8] ArUco: a minimal library for augmented reality applications based on opencv., 2015. URL: http://www.uco.es/investiga/grupos/ava/node/26.
[9] Matevž Bošnak, Drago Matko, and Sašo Blažič. Quadrocopter hovering using position-estimation information from inertial sensors and a high-delay video system. *Journal of Intelligent & Robotic Systems*, 67(1):43–60, 2012.
[10] Augmented Reality ARToolkit Patternmaker, 2016. URL: http://www.cs.utah.edu/gdc/projects/augmentedreality/.
[11] Harik El Houssein Chouaib, Guérin François, Guinand Frédéric, Brethé Jean-François, Pelvillain Hervé, and Zentout Adel. Vision based target tracking using an unmanned aerial vehicle. In *IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO 2015)*, page 6, Lyon, France. 2015. URL: https://hal-insu.archives-ouvertes.fr/UNIV-LEHAVRE/hal-01170117v1.
[12] Miguel A Olivares-Mendez, Ivan Mondragon, Pascual Campoy Cervera, Luis Mejias, and Carol Martinez. Aerial object following using visual fuzzy servoing. In *Proceedings of the 1st Workshop on Research, Development and Education on Unmanned Aerial Systems (RED-UAS 2011)*, pages 61–70. Centro Avanzado de Tecnologías Aeroespaciales (CATEC), 2011.
[13] Gianluca Antonelli, Stefano Chiaverini, and Giuseppe Fusco. A fuzzy-logic-based approach for mobile robot path tracking. *IEEE Transactions on Fuzzy Systems*, 15(2):211–221, 2007.
[14] Patrick Benavidez and Mo Jamshidi. Mobile robot navigation and target tracking system. In *System of Systems Engineering (SoSE), 2011 6th International Conference on*, pages 299–304. IEEE, 2011.
[15] Andrea Cesetti, Emanuele Frontoni, Adriano Mancini, Primo Zingaretti, and Sauro Longhi. A vision-based guidance system for uav navigation and safe landing using natural landmarks. In *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*, pages 233–257. Springer, 2009.
[16] Jeffrey W Tweedale. Fuzzy control loop in an autonomous landing system for unmanned air vehicles. In *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*, pages 1–8. IEEE, 2012.
[17] Joseph J LaViola Jr. "an experiment comparing double exponential smoothing and kalman filter-based predictive tracking algorithms". In *Virtual Reality, 2003. Proceedings. IEEE*, pages 283–284. IEEE, 2003.
[18] François Guerin, Simon G. Fabri, and Marvin K. Bugeja. "double exponential smoothing for predictive vision based target tracking of a wheeled mobile robot". In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 3535–3540, Florence, Italy, December 2013.
[19] John H Lilly. *Fuzzy control and identification*. John Wiley & Sons, 2011.
[20] Hao Ying. *Fuzzy control and modeling: analytical foundations and applications*. Wiley-IEEE Press, 2000.
[21] robotnik. Summit xl, 2016. URL: http://www.robotnik.eu/mobile-robots/summit-xl/.