2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Visual Servoing of UAV Using Cuboid Model with Simultaneous Tracking of Multiple Planar Faces

Manlio Barajas*[1], José Pablo Dávalos-Viveros[1], Salvador García-Lumbreras[1] and J.L. Gordillo[1]

Abstract— **Pose estimation is a key component for robot navigation. An Unmanned Aerial Vehicle (UAV) that is instructed to reach certain location requires a way of measuring its pose. This article presents a method for UAV visual servoing that uses the 3D pose of the drone as controller feedback. A remote monocular camera observes the tracked UAV while it moves and rotates in a 3D space. Pose is obtained from a 3D tracking process based on a cuboid model. In particular, a simultaneous face tracking strategy where 3D pose estimations from different faces are combined is introduced. Face combination was validated using a robotic arm with a cuboid at the final joint. For the UAV control, hover and path following tasks were tested. Results show that the proposed method correctly handles changes in pose, even though no face is always visible. Also, the UAV maintained a low speed in order to satisfy the small inter-frame displacement constraint imposed by visual tracking algorithm.**

I. INTRODUCTION

A localization method and a control strategy are determinant elements in successful autonomous navigation tasks for unmanned vehicles. Visual 3D pose estimation has been an important topic in literature because cameras have shown to be a reliable source of environment information. Common pose estimation approaches depend on depth information by means of stereo setups [1], [2] or laser range data [3].

With the advent of low cost drones, Unmanned Aerial Vehicle (UAV) control has become an active research topic. In particular, visual servoing is a common control approach to this problem since there are well known methods for achieving position estimation at image level. For example, Image Based Visual Servoing (IBVS) algorithms presented in [4], [5], [6], [7] are based on the tracking of a landmark placed on the floor. Others utilize IBVS for tracking a moving target and following it (physically) [8], [9]. Some approaches are on the context of Simultaneous Localization and Mapping (SLAM) where multiple sensor information is fused for more robust pose estimation [10], [11], [12]. Stereo vision is also used for pose estimation [13], [14]. These methods usually rely on data obtained from multiple sensors on the drone for a robust control system.

3D object tracking can be used to deliver 3D pose information to the vehicle controller. This may be accomplished by using full featured models [15], [16], [14] or by using approximated models [17]. While full featured models allow having a very exact model representation, this is not well suited in some cases, such as when real time performance is required. Approximated models can be used to overcome this constraint. This was validated at theoretical level in [18].

This article presents theoretical and implementation results on the visual tracking and servoing of an UAV using only visual plane tracking for 3D pose estimation. This research is based on the method presented in [18] for 3D pose tracking using individual planar faces of a cuboid shaped object. For the UAV visual servoing, the control system used is the one presented in [19]. In addition to those works, this article contributes to the state of the art as follows:

- Pose estimation is improved by tracking more than one face at a time on certain critical poses. Critical poses are those when a certain face of the cuboid has little visibility. As presented in [19], when the UAV rotates with respect to the camera, switching faces usually introduces perturbations to the system. This is related to the fact that homography decomposition is affected by sampling error, which is likely to increase as the area visible by the camera reduces.
- Full path following experiments are presented. In previous work [19], the stability of the system was strongly compromised because of face switching stage. For this reason, only hover experiments were reported.
- A single tracked element is used to achieve full path following. Since the camera is remote, the vehicle can perform more complex exercises without being restricted by marks on the floor or walls.

This article is divided as follows: section II will introduce related work to the problem of 3D tracking and UAV visual servoing. On section III, a method for 3D object tracking using multiple planar faces is reviewed. Section IV presents the control strategy used to achieve control over the UAV. Section V presents the experimental results and a comparison with other methods used for UAV control. Finally, in section VI conclusions are remarked.

II. RELATED WORK

In this work, a Position Based Visual Servoing (PBVS) system is used to control the 3D pose of an UAV.

For tracking and 3D pose estimation, our work relies on standard 2D plane tracking and homography decomposition for 3D reconstruction. The plane tracker used was introduced by Benhimane and Malis in [20].

3D Tracking in the context of image alignment has already been studied. The method presented by Cobzas et al. [21] uses standard 2D tracking over a 3D pose parameter space (6 parameters, one for each DoF). A drawback is that, when changing the parametrization to handle 3D pose changes,

[1]Center for Robotics and Intelligent Systems (CRIS), Tecnológico de Monterrey, Monterrey 64849, Mexico
*Corresponding author manlito@gmail.com

it results that even some movements that could be tracked without problems by tracking planes individually can produce failure. Another idea introduced by them is to apply constraints between planes to make tracking more robust.

On the same line of parametrizing directly in Euclidean space, Panin and Knoll [15] proposed a method for tracking objects that uses Mutual Information as similarity measure, instead of the common sum of squared differences (SSD), and perform a Levenberg-Marquardt optimization. Authors report performance of 2 *fps* because of the Mutual Information step. Moreover, the method requires a projection of a full CAD model on each iteration.

Visual servoing of aerial vehicles has been a thoroughly researched topic. For example, in [5], a pure image based visual servoing (IBVS) for landing and take-off process is developed. While this work has the advantage of not requiring a full 3D reconstruction, it can't be used for full object control (executing paths). In [6], a planar patch is also used for 3D pose estimation of the UAV. This is closer to our work, but it's restricted in space. In [10] a monocular SLAM system is used to eliminate odometry drift. This method proved to be very robust, but it requires the use of internal sensors like gyroscopes, accelerometers and cameras. In a similar manner, in [11] a SLAM algorithm tracks the pose of the on board camera and an on board PD controller is used for the attitude control. In [13] the full object pose is also controlled. However, this approach requires a stereo setup for 3D reconstruction and uses IMU data for rotational dynamics. A common pattern in these works is the use of on board cameras and sensors of the UAV. In our work, we used a single remote camera without relying on any data provided by the internal sensors

On the approach of remote sensing and control, in [14] a quadrotor is controlled using remote cameras and on board sensors. A binocular system with a previously known CAD model is used for pose estimation.

III. 3D TRACKING OF OBJECTS USING MULTIPLE PLANAR FACES

This article presents a method for Position Based Visual Servoing (PBVS) that uses as feedback the 3D pose (3D orientation and 3D position) of the UAV estimated using the 3D tracking method proposed in [18]. This method uses planar faces as visual tracking targets, over which standard plane tracking is done [22].

In the tracking method presented in [18], only one face of a cuboid is tracked at a time. A face selection criteria is used to determine which face is the optimal for tracking. Once a 2D homography between the reference template and the current image is obtained, homography decomposition is used to obtain a 3D reconstruction of that face. Next, a set of precomputed 3D transforms between the different faces of the model allow obtaining a full 3D reconstruction of the cuboid and the pose of the drone.

Here we propose to use more than once face at a time to improve pose estimation at critical poses. This occurs when the pose of the object with respect to the camera allows viewing and tracking individually more than 2 faces. Then, multiple poses are obtained which most be combined. A simple averaging rule is used.

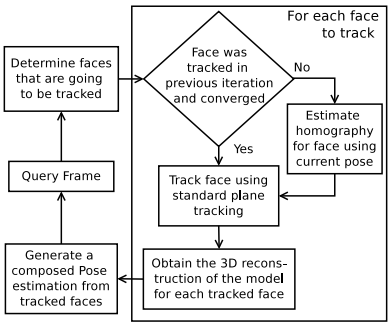

Fig. 1. Overview of of 3D tracking process using simultaneous tracking of planar faces. The number faces that are tracked simultaneously depends on the shape of the model and the number of visible faces.

Without the loss of generality, a cuboid model is used for tracking. Nevertheless, the method can be used with other polygon meshes of quadrilaterals, considering the relation between faces are known and that they are large enough to be tracked without image sampling issues. The full 3D pose tracking process is shown in Fig. 1.

The proposed method assumes the initial state of the target object is known. An initial state can be obtained using a single planar face, for which its initial homography should be available.

A. Cuboid Model

This work uses a cuboid model for tracking. A cuboid can be defined in terms of its 6 planar faces as follows:

$$C = \{\mathbf{P}_i \mid 1 \leq i \leq 6\} \tag{1}$$

Where $\mathbf{P}_i$ refers to the homogeneous transformation matrix for plane π_i from its current referential to camera referential. For all faces in the cuboid C there exist at least one transformation matrix ${}^{i}\mathbf{T}_k$ that maps face referential *k* to face referential *i*. This can be expressed as:

$$\forall \pi_i \in C \; \exists \; {}^{i}\mathbf{T}_k \mid \mathbf{P}_i = \mathbf{T}_k \mathbf{P}_k \tag{2}$$

In the case of the cuboid, these transforms can be estimated by assuming certain dimensions (up to scale factor) and then applying the corresponding transform.

B. Face Visibility

From the set of faces of cuboid C, only a subset of 1 up to 3 of the faces will be visible at anytime, assuming the cuboid is always visible by the camera. Moreover, there will be some cases in which, even if the face is visible, tracking it will not be possible due to image sampling (Fig. 2).

To determine the faces that are visible and suitable for tracking, a similar criteria to that used in [18] is used. That subset of faces is obtained as follows:

$$\nu = \{i \mid \angle \mathbf{cn}_i \geq 0 \wedge \angle \mathbf{cn}_i < \epsilon, \; \forall i \; : \; 1 \leq i \leq 6\} \tag{3}$$

Fig. 2. The UAV used for visual servoing. At the pose shown, two faces of the cuboid are visible. Anyway, only the face with 'suns' should be used, since the other has a small visible area.

Where $\mathbf{c}$ refers to the center of the tracked object relative to the camera coordinate system. $\mathbf{n}_i$ is a normal vector to plane i. Threshold ϵ sets the maximum allowed angle between vectors $\mathbf{c}$ and $\mathbf{n}_i$.

C. Plane Tracking of Planar Faces

The 3D pose tracking method employed is based on standard plane tracking. In particular, the Efficient Second Order Minimization (ESM) proposed by Malis [23] was used. This method has proven to provide higher convergence rate with less global error [23].

This method works by iteratively updating parameters $\mathbf{p} := \mathbf{p} \circ \mathbf{\Delta p}$ where $\mathbf{\Delta p}$ can be evaluated as:

$$\Delta \mathbf{p} \approx -2(\mathbf{J}(\mathbf{e}) + \mathbf{J}(\mathbf{p}_c))^{+}(\mathbf{s}(\mathbf{p}_c) - s(\mathbf{e})) \tag{4}$$

Where $\mathbf{p}$ are the current parameters, $\mathbf{J}$ is the Jacobian, as presented in [22], and $\mathbf{s}$ is the transformed current image. $\mathbf{e}$ is the identity parameter set, in this case $\mathbf{0}$. This method assumes the following homography parametrization:

$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \begin{pmatrix} 1+p_1 & p_3 & p_5 \\ p_2 & 1+p_4 & p_6 \\ p_7 & p_8 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{5}$$

D. Per Face 3D Object Reconstruction

The approach proposed in [18] is implemented to obtain a 3D pose of target object to feed the control loop. Real world dimensions of the object are provided in order to obtain a full Euclidean reconstruction.

In the 3D reconstruction stage, two main steps are done. First, the homography that is obtained using the ESM tracker is used to calculate its decomposition. Then, using transformations matrices, the center of the object is computed.

Homography decomposition deals with reconstructing the 3D pose of a planar surface given its projection on image plane. When working with plane tracking, usually only 8 parameters are used to accomplish tracking, which are mapped to the 3×3 homography transform, as shown in equation 5.

According to pinhole camera model, the transformation matrix of extrinsic parameters $\mathbf{P} = [\mathbf{R} \mid \mathbf{t}]$ is a matrix composed of rotation $\mathbf{R}$ and translation $\mathbf{t}$. In order for $\mathbf{P}$ to be a valid transformation, $\mathbf{R}$ must be an orthonormal basis.

The extrinsic camera parameters can be obtained as:

$$\mathbf{P} = [\mathbf{r_1}\ \mathbf{r_2}\ \mathbf{r_3}\ \mathbf{t}] \tag{6}$$

$$\mathbf{P} = \alpha \begin{bmatrix} \frac{h_{11}-c_x h_{31}}{f_x} & \frac{h_{12}-c_x h_{32}}{f_x} & \mathbf{r}_1 \times \mathbf{r}_2 & \frac{h_{13}-c_x}{fx} \\ \frac{h_{21}-c_y h_{31}}{f_y} & \frac{h_{22}-c_y h_{32}}{f_y} & \alpha & \frac{h_{23}-c_y}{fy} \\ h_{31} & h_{32} & & 1 \end{bmatrix} \tag{7}$$

Where h_{ij} are the elements of a homography matrix. c_x, c_y are camera center offset parameters. f_x and f_y are focal distances of the camera. α can be estimated as $1/||\mathbf{r_1}||$. In practice it can be necessary to apply an orthogonalization to the estimated $\mathbf{P}$, since the input homography may ignore error sources, such as noise, pixelation and camera distortion.

Homography decomposition must be done for each of the faces that are being tracked, as indicated in Fig. 1.

E. Pose Combination

When camera is calibrated manually, estimated intrinsic parameters may not be accurate. In such cases, homography decomposition will not be accurate either, since it depends on camera parameters. Also, it is well known that cameras may distort the image. In addition, as the tracked plane goes farther from the camera, image sampling errors will also affect decomposition accuracy. For these reason, depending on the face that is being tracked, different pose estimations will be obtained.

The objective of pose combination is to merge the available 3D poses from tracked faces. The combination is done separating rotation and translation. As a preliminary step of the combination, plane tracking is checked for convergence. This done using the average squared error of the plane tracking stage. Average squared error $\overline{e}$ is defined as:

$$\overline{e}(\mathbf{p}_c) = \frac{(\mathbf{s}(\mathbf{p}_c) - s(\mathbf{e}))^2}{l} \tag{8}$$

This equation is related to equation 4, since $s(\mathbf{p}_c) - s(\mathbf{e})$ is the *error image* at any given iteration. This error image is averaged using the number of pixels l of the template.

The combination of the rotation estimates is done by converting the available rotations to quaternions and then averaging. Since in quaternions $q_i = -q_i$ and poses should be close each other, the following assignment rule is used:

$$q_i = \begin{cases} f(i) & q(i) \cdot q_0 > 0 \\ -f(i) & \text{otherwise} \end{cases}, \forall i \in \nu \tag{9}$$

Where f is a function that converts rotation matrix estimated using face i to quaternions. From the two possible quaternion representations, this expression selects the one that has the signs that correctly average them. Finally, the averaged quaternion is normalized and returned to the matrix representation.

For the case of translation, element-wise average of translation estimations is used.

IV. UAV VISUAL SERVOING

The objective of this work is having an aerial vehicle to execute a given trajectory, provided a set of 3D poses that are part of the path. In the case of the implementation platform, possible manipulations are: *roll*, *pitch*, *yaw* and *vertical speed*. Roll and pitch control allow the UAV to move horizontally, coplanar to the floor (*xy* plane, as shown in Fig. 5). Yaw rotates the UAV in an axis perpendicular to the *xy* plane. Vertical speed is used to increase or decrease the height of the drone.

The *xy* position control was achieved using a cascade controller. At the parent level, a PD controller manipulated the speed in x and y at which the vehicle should move. In the nested loop, a PID controller manipulated the roll and pitch variables of the drone. Fig. 3 shows this controller.

Controller design is closely related to the visual method used to determine the pose of the UAV. This method relies on standard plane tracking which assumes small inter-frame displacement. Fast UAV motions can cause the tracker to diverge. This is important in the case of perturbations, where it's desired to return to the reference softly.

Integral component of the speed plays a key role when perturbations are present. In case of wind gusts, it will help to reduce the error by increasing the manipulation when the vehicle is having difficulty to move.

For the yaw and vertical speed, standard proportional controllers were used, since drone built-in orientation and height control is stable for tracking. Fig. 4 shows the proposed controller for yaw. A similar controller is also used for height control.

V. EXPERIMENTAL RESULTS

Experiments were designed to validate that the proposed combination strategy is able to improve UAV pose estimation for visual servoing. The pose is estimated visually from a cuboid shaped object that is installed on the top of the drone (see Fig. 5).

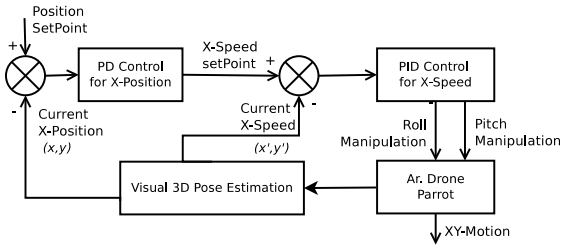

Fig. 3. Cascade controller controller for xy position.

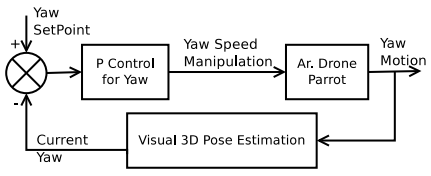

Fig. 4. Controller for yaw. A similar controller is used for height control.

TABLE I
INDIVIDUAL AND COMBINED ESTIMATION ON CRITICAL POSE

Pose	Measured	Face 0	Face 5	Combined
roll (deg)	45.0	48.5	42.4	45.4
pitch (deg)	0.0	-1.23	-0.5	-1.2
yaw (deg)	0.0	-5.6	-5.9	-5.6
x (mm)	25.0	27.4	26.9	27.2
y (mm)	700.0	693.9	695.3	694.6
z (mm)	0	-13.3	-13.5	-13.4

First, a validation exercise was carried out to verify how the combination of faces behave in a controlled setup. In this experiment, a cuboid shaped object mounted on the last join of a robotic arm was used. Robot odometry was used as a ground truth measure.

Finally, experiments to test the visual servoing strategy using the UAV were run. In these experiments, the UAV rotated and translated with respect to the camera in order to verify that the combination of poses can be used in a real set-up.

A. Controlled Setup for Cuboid Tracking with Simultaneous Tracking of Faces

For this exercise, a cuboid shaped object was installed at the end joint of a robotic arm *CRS F3*. Then, the robot was instructed to rotate 360° over that joint. A calibrated *SONY EVI-D30* VGA camera was used as video source. Fig. 6 shows results from this experiment.

During this experiment, the robot was instructed to stop at critical orientations to evaluate how the combination averaged results. Table I shows a sample for one of these critical poses.

B. UAV Visual Servoing Experiments

The UAV used for visual servoing was a *Parrot Ar.Drone 1.0* quadrotor. At the top of it, a cuboid shaped object was installed. This cuboid was used for 3D tracking and pose estimation. Control algorithms run from a *Intel Xeon* E5-1650 processor workstation, which manipulated the drone through a WiFi link. The camera used was a *Microsoft LifeCam Studio* with a selected output of 1280×720 @ 30fps.

Hover stability was verified. Some wind gusts were applied to assess UAV response to perturbations and the speed

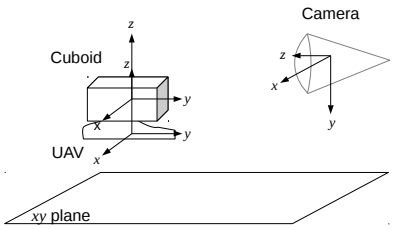

Fig. 5. A cuboid mounted on top of an UAV is viewed by a remote camera. A translation is applied from cuboid to UAV referential to obtain drone's pose.

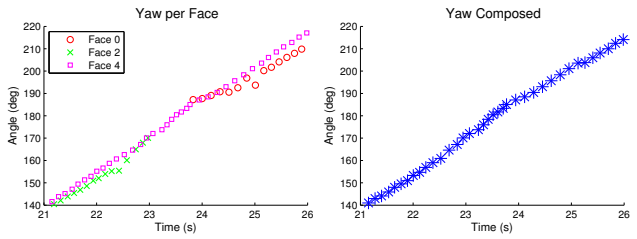

Fig. 6. Results for validation exercise using robotic arm. The robot was instructed to apply a rotation in the yaw axis of the cuboid. As figure on left shows, not all faces can be tracked all the time, and each face may have a different estimation. Plot on the right shows the combined result.

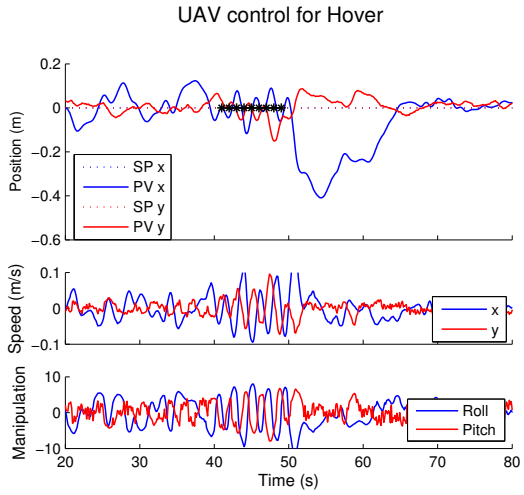

Fig. 7. Response of UAV control in hover. Some wind gusts were applied (denoted as * in top plot). It must be noted that the speed (middle plot) was kept low. This is to avoid divergence problems related to the plane tracker. Last plot shows the manipulation output for both roll and pitch.

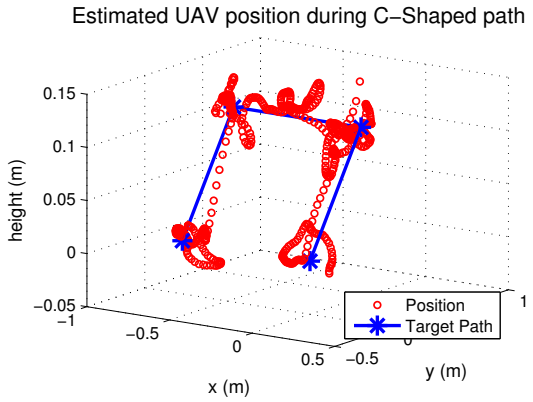

Fig. 8. Path executed by the drone in the C-Shaped experiment.

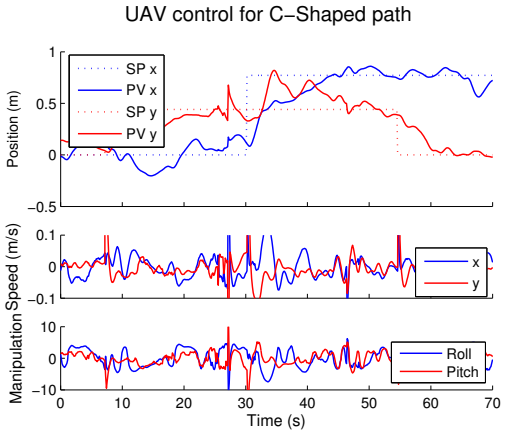

Fig. 9. UAV xy position control during the execution of C-Shaped path. Top plot shows how set points were applied and the obtained response. Some strong changes in pose estimation produced spikes in speed (middle). Manipulation is shown at bottom.

control (Fig. 7). The controller could accomplish the objective of responding softly to perturbations, regardless large differences against the reference.

The second control experiment consisted in the execution of a path with the shape of a 'C' with a certain slope, for a total of 4 setpoints. The target path and the executed one, as measured by the 3D pose tracking process, are shown in Fig. 8. Position control is shown in Fig. 9. Yaw and height control results are presented in Fig. 10.

C. Discussion

Results shown in Fig. 6 confirm that the proposed approach correctly handles the transition between faces, by smoothing the pose estimation which ultimately benefits the control stability. This allowed rotating without significantly affecting the control.

Another observation is that the UAV maintained a low speed, regardless of perturbations. This is important since when perturbations are retired, the vehicle still moves at constant speed. If speed is increased, paths are executed better but the probability of divergence of the plane tracking algorithm increases, specially during critical poses, when two or more faces are visible.

Performance during pose estimation stage was close to 20 fps. Nevertheless, communication with the drone was at 40 packets per second. The last estimated pose was used to calculate the control variables.

The response time is a significant difference between this work and others. In both [10], [11] a hover and a path following exercise were presented using a visual SLAM which used the on board IMU data for attitude control and localization. This allowed them to react faster to perturbations. For the presented approach, increasing the speeds compromises the stability of the visual tracking.

Methods [4], [5], [6], [7] assume a certain mark or template is always visible. This limits the possible navigation space, or requires multiple marks to be places along the way. Our approach instead uses a remote camera, which can have a larger coverage using a PTZ system.

VI. CONCLUSIONS

In this work, a visual only method for UAV control that uses the faces of a cuboid as reference was presented. This approach differ from others since a remote camera is used to track the 3D pose of the UAV, instead of using on board

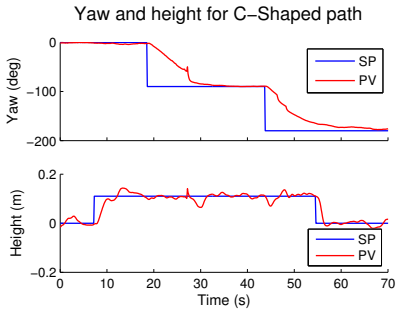

Fig. 10. Yaw (top) and height (bottom) control of UAV during the execution of C-Shaped path. Parrot drone has a stable built in control of these variables, which allowed to simplify the control over them by using only proportional control.

cameras. The choice of the cuboid is related to the fact that, if using only a plane, there will be configurations in which it won't be visible to the camera.

A validation experiment confirmed that a way of transitioning smoothly between faces is needed because of errors in the homography decomposition process. Also, since plane trackers may diverge, it was useful to define a criteria for determining if any given face should be tracked.

Regarding UAV control, we have validated this approach by executing paths that are in the three dimensional space. Since we required to have a low speed, a cascade controller was implemented to have a fine speed control. Unfortunately it had the side effect of being slow to reach steady state.

At the current state and similar to other works, navigation is limited to small areas. This is closely related to image sampling errors that increase significantly when the vehicle is far from the camera.

An advantage of having a remote camera is that it doesn't have image stabilization issues and that it can track multiple vehicles at the same time. In posterior work, multiple vehicle tracking and additional face combination criteria will be evaluated. Another area of opportunity for this research is the fusion of data from drone IMU and other sensors with that of the vision. This could help improving the pose estimation during face combination stage.

ACKNOWLEDGMENT

Authors want to acknowledge Consejo Nacional de Ciencia y Tecnología (CONACyT) and e-Robots Research Chair from Tecnológico de Monterrey, Monterrey for supporting this research.

REFERENCES

[1] Tonko, M. and H.-H. Nagel, "Model-Based Stereo-Tracking of Non-Polyhedral Objects for Automatic Disassembly Experiments," *International Journal of Computer Vision*, vol. 37, no. 1, pp. 99–118, 2000.

[2] R. Munoz-Salinas, E. Aguirre, and M. Garcia-Silvente, "People detection and tracking using stereo vision and color," *Image and Vision Computing*, vol. 25, no. 6, pp. 995–1007, 2007.

[3] X. Shao, H. Zhao, and K. Nakamura, "Detection and tracking of multiple pedestrians by using laser range scanners," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, no. 1, 2007, pp. 2174–2179.

[4] O. Bourquardez, R. Mahony, N. Guenard, F. Chaumette, T. Hamel, and L. Eck, "Image-based visual servo control of the translation kinematics of a quadrotor aerial vehicle," vol. 25, no. 3, 2009, pp. 743–749.

[5] D. Lee, T. Ryan, and H. Kim, "Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May, pp. 971–976.

[6] I. Mondragon, P. Campoy, C. Martinez, and M. Olivares-Mendez, "3D pose estimation based on planar object tracking for UAVs control," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 35–41.

[7] R. Ozawa and F. Chaumette, "Dynamic visual servoing with image moments for a quadrotor using a virtual spring approach," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 5670–5676.

[8] V. Chitrakaran, D. Dawson, H. Kannan, and M. Feemster, "Vision assisted autonomous path following for unmanned aerial vehicles," in *Decision and Control, 2006 45th IEEE Conference on*, 2006, pp. 63–68.

[9] I. Mondragon, P. Campoy, M. Olivares-Mendez, and C. Martinez, "3D object following based on visual information for Unmanned Aerial Vehicles," in *Robotics Symposium, 2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications (LARC)*, 2011, pp. 1–7.

[10] J. Engel, J. Sturm, and D. Cremers, "Camera-based navigation of a low-cost quadrocopter," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 2815–2821.

[11] M. Blosch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based mav navigation in unknown and unstructured environments," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 21–28.

[12] M. Achtelik, M. Achtelik, S. Weiss, and R. Siegwart, "Onboard imu and monocular vision based control for mavs in unknown in- and outdoor environments," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 3056–3063.

[13] S. Salazar, H. Romero, J. Gomez, and R. Lozano, "Real-time stereo visual servoing control of an uav having eight-rotors," in *Electrical Engineering, Computing Science and Automatic Control,CCE,2009 6th International Conference on*, Jan., pp. 1–11.

[14] S. Klose, J. Wang, M. Achtelik, G. Panin, F. Holzapfel, and A. Knoll, "Markerless, vision-assisted flight control of a quadrocopter," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, 2010, pp. 5712–5717.

[15] G. Panin and A. Knoll, "Mutual Information-Based 3D Object Tracking," *International Journal of Computer Vision*, vol. 78, no. 1, pp. 107–118, 2007.

[16] J. A. Brown and D. W. Capson, "A Framework for 3D Model-Based Visual Tracking Using a GPU-Accelerated Particle Filter." *IEEE transactions on visualization and computer graphics*, vol. 18, no. 1, pp. 68–80, Feb. 2011.

[17] M. Manz and T. Luettel, "Monocular model-based 3D vehicle tracking for autonomous vehicles in unstructured environment," *(ICRA), 2011 IEEE*, pp. 2465–2471, 2011.

[18] M. Barajas, J. Esparza, and J. Gordillo, "Towards Automatic 3D Pose Tracking through Polygon Mesh Approximation," in *Advances in Artificial Intelligence IBERAMIA 2012*, ser. Lecture Notes in Computer Science, J. Pavn, N. Duque-Mndez, and R. Fuentes-Fernndez, Eds. Springer Berlin Heidelberg, 2012, vol. 7637, pp. 531–540.

[19] M. Barajas, J. Dávalos-Viveros, and J. Gordillo, "3D Tracking and Control of UAV using Planar Faces and Monocular Camera," in *5th Mexican Conference on Pattern Recognition MCPR 2013*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, in-press.

[20] S. Benhimane and E. Malis, "Homography-based 2D Visual Tracking and Servoing," *The International Journal of Robotics Research*, vol. 26, no. 7, pp. 661–676, July 2007.

[21] D. Cobzas, M. Jagersand, and P. Sturm, "3D SSD tracking with estimated 3D planes ," *Image and Vision Computing*, vol. 27, no. 12, pp. 69 – 79, 2009.

[22] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221 – 255, March 2004.

[23] E. Malis, "Improving vision-based control using efficient second-order minimization techniques," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 2, 26-May 1,, pp. 1843–1848 Vol.2.