Vision-based Servo Control of a Quadrotor Air Vehicle

Zehra Ceren, Erdinç Altuğ

Abstract—**Unmanned aerial vehicles (UAVs) are seeing more widespread use in military, scientific, and civilian sectors in recent years. This study presents algorithms for the visual-servo control of an UAV. The helicopter has been stabilized with visual information through the control loop. Unlike previous study that use pose estimation approach which is time consuming and subject to various errors, the visual-servo control is more reliable and fast. The model involves the camera speed and visual sensor blocks. Visual sensor block consists of the camera model and feature extraction blocks. Various simulations are developed on MATLAB, in which the quadrotor aerial vehicle has been visual-servo controlled. In order to show the effectiveness of the algorithms, experiments were performed on a model UAV which suggest successful performance.**

I. Introduction

The visual-servoing from its simple forms to today's more advanced real-time versions has emerged considerably within the last 30 years. The need of increased flexibility and advanced task requirements promoted the use of vision in the control loop in robotics.

The visual-servo control uses the features on the image plane and servo controls them to the goal position [1]. Typically point based features are used which are centers of some features or the coordinates of the edges. Various studies have been performed on eye-at-hand systems in which a camera is placed on the end-effector of a manipulator to guide it to the desired location [2, 3].

The use of visual-servoing on aerial robots is very natural. The visual sensors are already available on various unmanned aerial vehicle (UAV) platforms to obtain remote visual information. This visual information can also be used for the control loop instead of more complex alternatives such as GPS or inertial navigation systems.

Recently various studies are performed on visual-servo control of UAVs, such as four-rotor aerial vehicles [4, 5], helicopters [6, 7], blimps [8], and aircrafts [9]. The navigation and control strategies for an aircraft that can fly towards to a known object using vision only for the feedback loop has been presented in [10]. The control methods to stabilize a four-rotor helicopter with vision as the main sensor have been presented in [4]. In this study, pose of the helicopter has been estimated and this information was used to control the vehicle. Vision based autonomous flying has been studied in [11, 12].

Vision based servo control has two feedback loops, inner and outer loop. Inner loop has lower level controls to control either the velocity of the robot [13] or the velocity of the camera [14]. On the other hand, the outer loop is a feedback loop which controls the error vector with the vision based controller.

In this study a simulation model and experiments for the visual-servo control of an unmanned quadrotor aerial vehicle is presented. Firstly, dynamical model of a four-rotor helicopter is developed in simulation [19]. The helicopter model has been stabilized with visual information through the control loop. Unlike previous study that use pose estimation approach which is time consuming and subject to various errors, the visual-servo control is more reliable and fast. The proposed system's block diagram is presented in Figure 1. The block diagram consists of a helicopter block, a feature extraction block, a camera and an image based control block. The camera is located under the helicopter and it is looking at the ground. Captured images are processed with a feature extraction routine to determine the features on the scene. Black blobs of various sizes were selected for simplicity and real-time performance need in experiments. The purpose of the feature extractor is to locate the center of these blobs and transfer eight image plane parameters of the four blobs to the image-based controller. The perspective projection and camera model are combined for the visual sensor. Using the helicopter model and the visual sensor model an open-loop visual servo controller is built. The vision-based controller is selected as a proportional derivative (PD) controller due to the simplicity. This controller will then drive the image coordinates of the features to the desired positions by first estimating the desired helicopter velocity and then driving the helicopter to this desired velocity by controlling spinning speeds of the four motors of the helicopter.

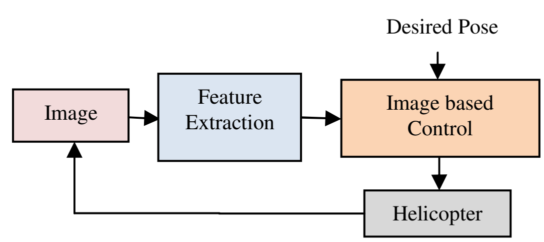

Fig. 1. The use of visual servo control for helicopter stabilization.

This work was supported in part by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant 107E211.

Z. Ceren is with the system dynamics and control graduate program at the İstanbul Technical University, İstanbul, TURKEY. (e-mail: cerenze@hotmail.com).

E. Altuğ is with the Mechanical Engineering Department, İstanbul Technical University, Istanbul, TURKEY (e-mail: altuger@itu.edu.tr).

II. THE CONTROLLER

The control architecture and image based visual servoing system refers to a mini quadrotor vehicle, a PTZ camera which is fixed on the quadrotor and the object of four point. It is assumed that the object always be seen in the camera field of view. The representation of closed loop visual servoing is shown on Figure 2.

The reference feature vector $\boldsymbol{f}^*$ is determined by using the image of object by considering the desired pose of the camera. The current feature vector $\boldsymbol{f}$ is obtained by using the image processing library. The error $\boldsymbol{e}$ is the variance between the reference feature vector $\boldsymbol{f}^*$ and current feature vector $\boldsymbol{f}$. The image based visual servoing control structure generates the control signal $\boldsymbol{v}_c^*$ by using the information about the error and the variance. Reference velocity $\boldsymbol{v}_c^*$ is a 6x1 vector as consist of linear and angular velocities of the camera. In other words, $\boldsymbol{v}_c^*$ is the required camera velocity information for performing the desired task. The aim of image based visual servo control is to generate a reference velocity vector $\boldsymbol{v}_c^*$ which minimizes the error. In this paper, a proportional image based controller is designed. Now we can see in Fig. 2 that the visual servoing system is a Multi-input Multi-Output system. The components of reference velocity $\boldsymbol{v}_c^*$ are the inputs and the coordinates of features $\boldsymbol{f}$ are the outputs of the system.

We can consider the system in two parts. The first one is the velocity controlled robot and the second one is the camera, image acquisition and processing. The inputs of helicopter model are the components of helicopter state vector $\boldsymbol{x}_h$, the reference pitch and roll angles θ^*, $\emptyset^*$, and the components of reference velocity vector $\boldsymbol{v}_h^*$. The helicopter state vector $\boldsymbol{x}_h$ is the vehicle's own state vector. The components of 6x1 $\boldsymbol{x}_h$ state vector is the translation and rotation of the quadrotor around the x, y, z axes respectively. In respect to the other helicopter inputs, how the reference pitch and roll angles are estimated is explained in the Helicopter Control section. The last input of helicopter is reference helicopter velocity vector $\boldsymbol{v}_h^*$. $\boldsymbol{v}_h^*$ is the required helicopter velocity information for performing the desired task. This vector is composed from reference camera velocity. In other words, camera reference velocity vector $\boldsymbol{v}_c^*$ is transformed to helicopter reference velocity vector $\boldsymbol{v}_h^*$ by using the pose of the camera related to the helicopter $\boldsymbol{T}_c^h$.

The altitude, roll, pitch, yaw controller were designed for helicopter. For clarity, each of the controllers is represented by four little boxes in Figure 2. The output of helicopter model is helicopter state vector $\boldsymbol{x}_h$. This 6x1 vector is feedback to the visual sensor model in a form of 4x4 homogeneous transformation matrix $\boldsymbol{T}_h^0$, by using transformation matrix $\mathbf{T}$ in Eq. 5.

III. VISUAL SERVO CONTROL SYSTEM

It is considered from Figure 2, the visual servoing control system is composed of the helicopter, camera and the object. Open loop visual servoing system model can be divided to three sub models. These are helicopter controller model, visual sensor and image based controller:

A. *Control of Helicopter*

The helicopter model used in this paper is explained in a detailed manner in [4, 15]. The helicopter controllers have four input commands as U_1, U_2, U_3, U_4. U_1 represents the translation around the z axis. U_2 represents the rotation around the y axis (roll angle). U_3 represents the rotation around the x axis (pitch angle). Finally, U_4 represents the rotation around the z axis (yaw angle).

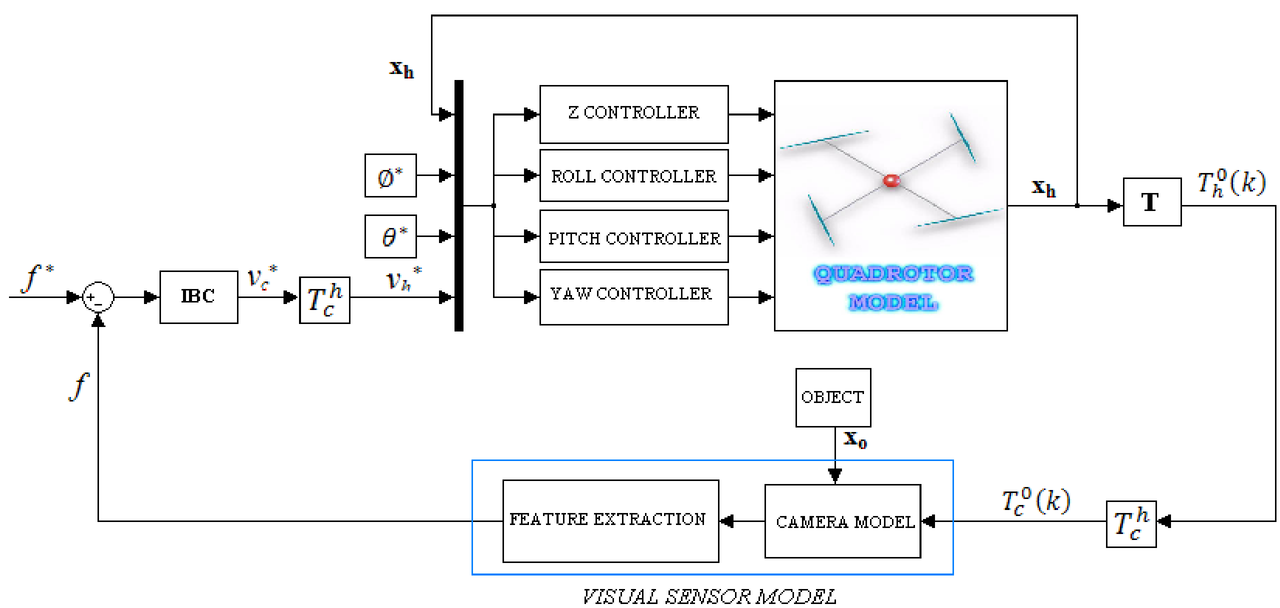

Fig. 2. The control system.

In this study, Proportional-Derivative (PD) controllers are designed to control the helicopter. This is because that the control algorithm can be obtained from the helicopter model and this algorithm makes the system exponentially stable as explained in [15].

Altitude Control:

For the altitude control of the helicopter Eq. 1 is used [16].

$$U_1 = \frac{mg}{cos\theta cos\emptyset} + \frac{m[kd_z(\dot{z} - \dot{z}^*)]}{cos\theta cos\emptyset} \quad (1)$$

where, $\dot{\boldsymbol{z}}^*$ is the reference linear velocity value around the z axis which is the third component of helicopter reference velocity vector vh*.

Translation Control:

It is necessary to control the pitch and roll angles for controlling the translations around the x and y axis. Therefore, for translation around x axis, reference pitch angle and angular rate of pitch angle ($\boldsymbol{\theta}^*, \dot{\boldsymbol{\theta}}^*$) are demanded. In the same way, for translation around y axis, reference roll angle and angular rate of roll angle ($\emptyset^*$, $\dot{\emptyset}^*$) are demanded. While the angular rates are determined from $\boldsymbol{v_h}$* vector (4th and 5th components), the angles are determined by using Eq. 2:

$$\begin{aligned} \emptyset^* &= arcsin[kd_y(\dot{y}^* - \dot{y})] \\ \theta^* &= arcsin[kd_x(\dot{x}^* - \dot{x})] \end{aligned} \quad (2)$$

In this way the controllers are designed as:

$$\begin{aligned} U_2 &= kp_\emptyset(\emptyset^* - \emptyset) - kd_\emptyset\dot{\emptyset} \\ U_3 &= kp_\theta(\theta^* - \theta) - kd_\theta\dot{\theta} \end{aligned} \quad (3)$$

Yaw Control:

Desired input signal for the yaw control of the helicopter is presented in Eq. 4:

$$U_4 = kd_\varphi(\dot{\psi}^* - \dot{\psi}) \quad (4)$$

The reference angular rate $\dot{\psi}^*$ is the sixth component of the $\boldsymbol{v_h}^*$ vector.

The controlled helicopter produces the position information as the output. The position of the helicopter is consisting of the positions and orientations around x, y, z axes respectively. That is to say the position of helicopter is expressed in a 6x1 vector form as $\boldsymbol{x_h} = \{x, y, z, \emptyset, \theta, \psi\}$. But because of its structure, the camera model cannot accept the $\boldsymbol{x_h}$ in 6x1 vector form. Therefore, before the state vector $\boldsymbol{x_h}$ is feedback to the camera it is converted to a 4x4 homogeneous transformation matrix form ($\boldsymbol{T_h^0}$) by using Eq 5.

$$\mathbf{T} = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\emptyset - s\psi c\emptyset & c\psi s\theta c\emptyset + s\psi s\emptyset & x \\ s\psi c\theta & s\psi s\theta s\emptyset + c\psi c\emptyset & s\psi s\theta c\emptyset - c\psi s\emptyset & y \\ -s\theta & c\theta s\emptyset & c\theta c\emptyset & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

B. Visual Sensor

Typically, the visual sensor consists of a camera and image processing block. In the simulation the object was defined as 3x1 vectors of coordinates related to earth for each points by 'polyhedra' command in MATLAB [17]. To characterize the object four feature points were selected, being defined as:

$$f = [f_1^T, \dots, f_m^T]^T \quad f_i = [f_{xi}, f_{yi}]^T \quad (6)$$

where $\boldsymbol{f_i}$ is the vector with the i-th feature coordinates in the image plane.

The camera was modeled by using the positions and orientations of the camera and the object ($\boldsymbol{x_c}, \boldsymbol{x_o}$). The image processing block is modeled in [7] as:

$$f = g\big(i(x_c, x_o)\big) \quad (7)$$

where $\boldsymbol{i}$ is the mapping function and $\boldsymbol{g}$ is a function that models the feature extracting algorithm.

For a 3-D point with coordinates X = (*X, Y ,Z*) in the camera frame, which projects in the image as a 2-D point with coordinates x = (*x, y*), we have in Eq. 8:

$$\begin{aligned} x &= \frac{X}{Z} = (u - c_u)/f_u \\ y &= \frac{Y}{Z} = (v - c_v)/f_v \end{aligned} \quad (8)$$

where m = (*u, v*) gives the coordinates of the image point expressed in pixel units, and a = ($\boldsymbol{c_u}, \boldsymbol{c_v}, \boldsymbol{f_u}, \boldsymbol{f_v}$) is the set of camera intrinsic parameters. $\boldsymbol{c_u}$ and $\boldsymbol{c_v}$ are the coordinates of the principal point, $\boldsymbol{f_u}$ and $\boldsymbol{f_v}$ are the focal lengths. In this study, the image plane coordinates of the points are taken as feature vectors as shown in Eq. 9.

$$\boldsymbol{f} = \mathrm{x} = (x, y) \quad (9)$$

The details of imaging geometry and perspective projection can be found in many computer vision texts.

To develop the visual sensor model, first the frames are defined. The helicopter frame is $\mathbf{R_h}$, the camera frame is $\mathbf{R_c}$ and the object frame is $\mathbf{R_o}$ as shown in Figure 3.

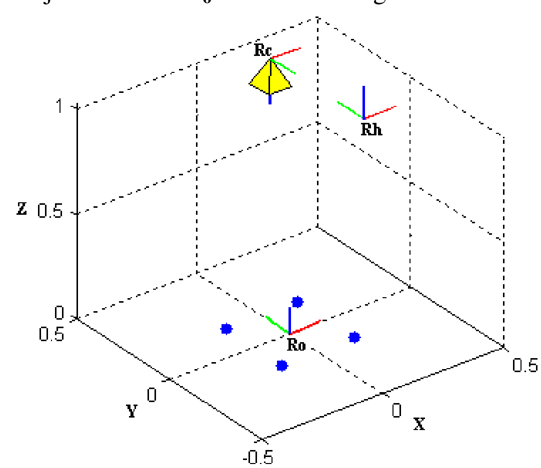

Fig. 3. The axes of the camera, object and helicopter.

$\boldsymbol{T_h^0}$, $\boldsymbol{T_o^0}$ are the homogeneous transformation matrices of the helicopter and object, respectively. They are defined related to earth. The axis of earth is defined by the subscript of '**0**'. In this paper, the start and desired positions of the helicopter are considered known. Also the first and desired position of the camera is obtained from the known positions of helicopter by using $\boldsymbol{T_c^h}$ matrix. The homogeneous transformation matrix $\boldsymbol{T_c^h}$ is the position of the camera related to helicopter. In this study, $\boldsymbol{T_c^h}$ is constant because of the camera is fixed with respect to the helicopter.

During the simulation, the helicopter homogeneous transformation matrix $\boldsymbol{T_h^0(k)}$ at the current discrete moment k is determined from the output vector $\boldsymbol{x_h}$ by using Eq. 5. Then, $\boldsymbol{T_h^0(k)}$ is converted to the camera homogeneous transformation matrix $\boldsymbol{T_c^0(k)}$ by using the $\boldsymbol{T_c^h}$ matrix. Thus, the input signal of the camera model is obtained for the simulation. In Figure 4, the visual sensor input and output signals are shown clearly.

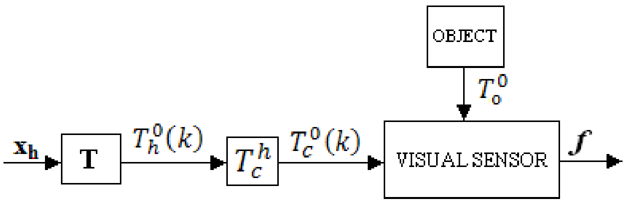

Fig. 4. Simulation inputs and outputs for visual sensor.

Finally, feature vector $\boldsymbol{f}$ is obtained by feature extraction methods.

C. Image Based Controller Model

The aim of image based controller is to minimize the error $\boldsymbol{e}$. The error is determined in Eq. 10.

$$e(t) = f^* - f(t) \tag{10}$$

where the desired feature $\boldsymbol{f}^*$ represents the point features obtained at the final position of the helicopter after the motion done for reaching a target pose. Desired (reference) feature vector $\boldsymbol{f}^*$ is obtained with using information of object and the desired position of camera in Eq. 7. In this paper motionless target object and constant target position was considered. Therefore $\boldsymbol{f}^*$ is constant. The changes on $\boldsymbol{f}$ only depend on the motion of camera.

The $\boldsymbol{v_c}^*$ is the output of image based controller and it is tend to decrease the error exponentially. The relation between the variance in time of error and the reference camera velocity is developed in Eq. 11 [1].

$$\dot{e} = L_e v_c^* \tag{11}$$

Thus the obtained proportional control algorithm is derived in [12].

$$v_c^* = -\lambda {L_e}^{-1} e(t) \tag{12}$$

where $L = [L_1^T, \ldots, L_m^T\,]$ is the image Jacobian with $\boldsymbol{L_i}$ defined as:

$$L_i = \begin{bmatrix} -\frac{1}{z} & 0 & \frac{x}{z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{z} & \frac{y}{z} & 1+y^2 & -xy & -x \end{bmatrix} \tag{13}$$

In this paper feature interaction matrix $\boldsymbol{L_e}$ is obtained by using MATLAB Visual Servoing Toolbox [18]. In control algorithm pseudo inverse of $\boldsymbol{L}$ is used.

IV. SIMULATIONS

The simulation model was tested using a visual servoing toolbox [18] in MATLAB. The quadrotor has the eye in hand configuration and the object is motionless and it is located at the ground. The visual controller is a proportional one and the controller constant; lambda has a value of 2. It is desired from the helicopter to move 2 meter down and 10 degree yaw angle in negative direction with the given target position. In this study, all lengths are defined in meter and angles are defined in radian. For clarity, the angle values in the simulation results are shown in degrees.

During the motion from the start position to final position, the trajectory of helicopter is shown in Figure 5. It can be seen that the helicopter keeps the pitch and roll angles stable and reaches to the desired yaw angle. At the same time it provides the desired descent. The motion along x and y axes during the simulation is negligible.

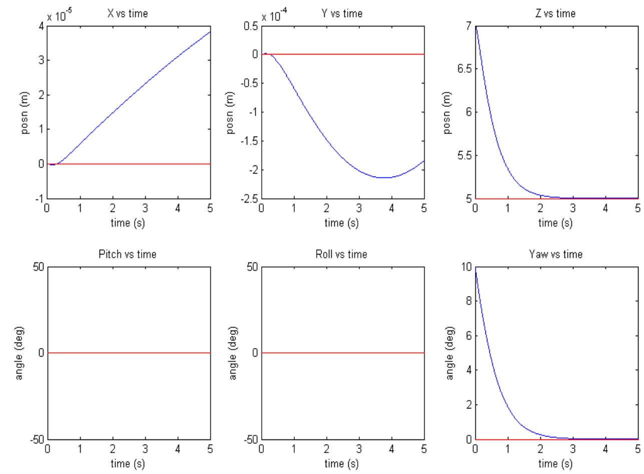

Fig. 5. Pose of the helicopter

In Figure 6 the helicopter velocities are shown during the motion. In this figure $\boldsymbol{v_x}$, $\boldsymbol{v_y}$, $\boldsymbol{v_z}$ represent the linear velocities of the quadrotor and $\boldsymbol{w_x}$, $\boldsymbol{w_y}$, $\boldsymbol{w_z}$ represent the angular velocities of the helicopter. It is clear that all the velocities reaches zero at the final position.

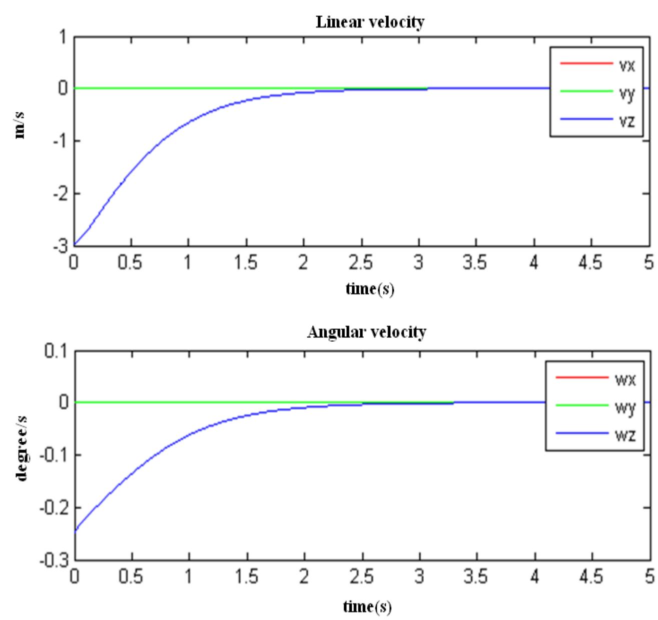

Fig. 6. Linear and angular velocities of the helicopter

In Figure 7 the motion of the camera on the helicopter's presented. The red axis is the x-axis, the green axis is the y-axis and the blue line represents the z-axis of the camera as it moves. The yellow object represents the camera.

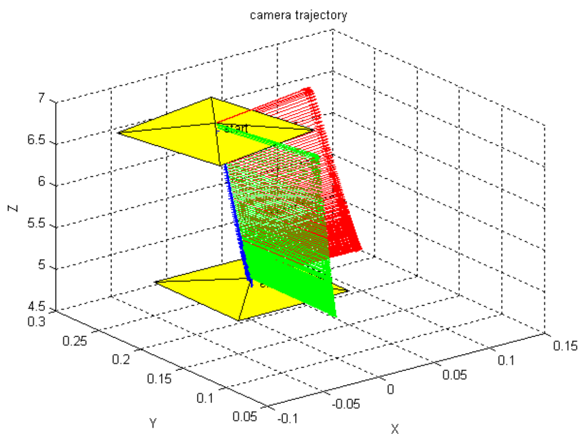

Fig. 7. Motion of the camera

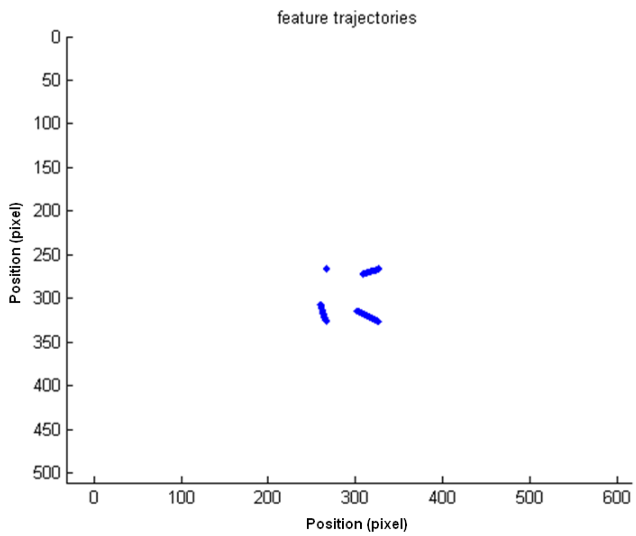

Fig. 8. The motion of the features

The motion of the features on the image plane during the simulation is presented in Figure 8. Since the helicopter is moving downwards the features are moving outwards the image center as expected. Due to the yaw motion of the helicopter, the features turn as they move.

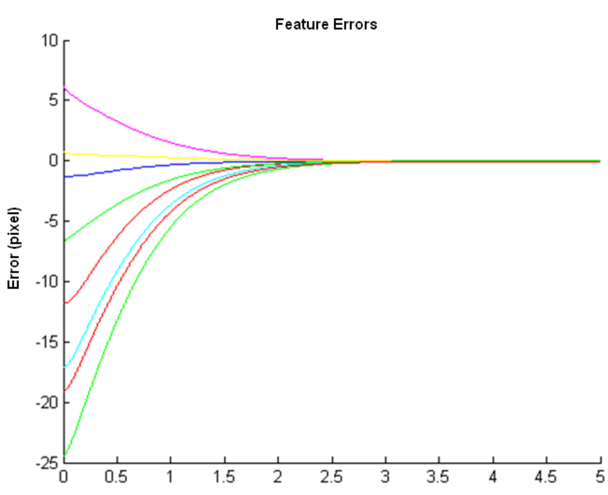

Fig. 9. Surface errors

The change of surface errors with time is presented in Figure 9. All the errors converge to zero as the simulation continues.

Various simulations are developed on MATLAB, in which the quadrotor aerial vehicle has been visual-servo controlled successfully. PD helicopter controllers successfully generated control signals to move the helicopter towards the desired position as generated by the image based controller.

V. EXPERIMENTS

A custom designed experimental test stand shown in Figure 10 is used to perform secure experiments. The test stand allows the helicopter to perform yaw motions freely, allows up to 2 meters of altitude, as well as up to ±20° roll and pitch motion. The experiment system consists of a model quadrotor helicopter, a test stand, a pan/tit/zoom camera, a catadioptric camera to be used in the future researches, and an IMU sensor. A Core2Quad 2.40 Ghz processor desktop computer with 3 GBs RAM on Windows XP that has a dual frame-grabber has been used. Algorithms were developed using Matrox Imaging Library 8.0 on C++ [20]. A Sony pan/tilt/zoom camera is directed to a stationary ground target. Captured images are processed with a feature extraction routine to determine the black blob features on the scene. Black blobs of various sizes were selected for simplicity and real-time performance.

Fig. 10. Experimental system

In order to show the effectiveness of the proposed algorithms an experiment of yaw motion under visual-servo control has been performed. The helicopter starts at 70 degree yaw angle and the goal is to reach 110 degree yaw angle under visual servo control.

The Euler angles of the helicopter during the experiment are presented in Figure 11. The helicopter reaches the desired yaw values as the roll and pitch angles are kept at zero degrees during the motion.

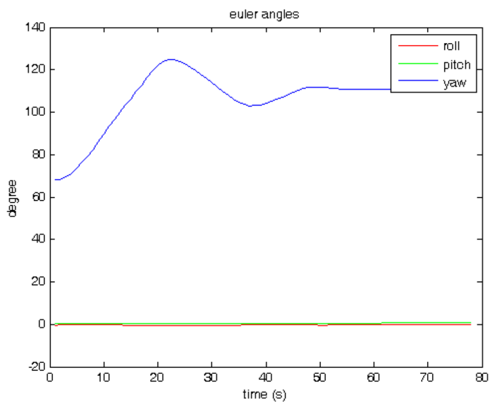

Fig. 11. The Euler angles of the helicopter during the experiment.

The linear and angular velocities during the experiment are presented in Figure 12. The desired angular velocity w_z which is related with the yaw motion approaches zero line as helicopter approaches the desired yaw angle.

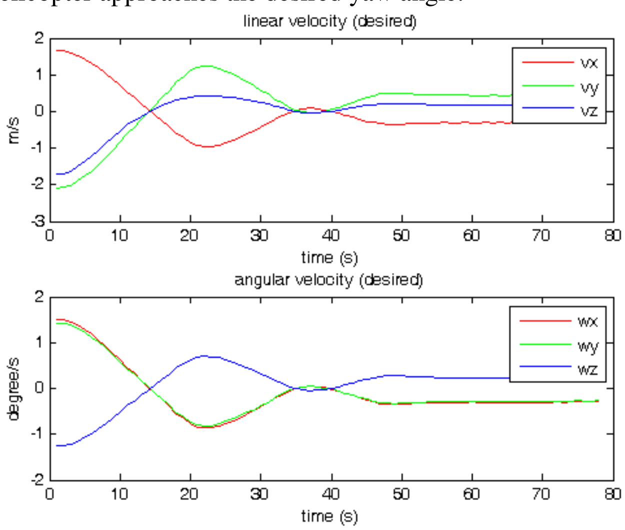

Fig. 12. Results of the yaw control experiment.

Initial experiments show that the system can determine the features, successfully estimate the desired velocity vector and guide the helicopter under visual-servo control.

VI. Conclusion

In this study, control algorithms have been developed for the visual-servo control of a quadrotor unmanned aerial vehicle. Visual information has been used solely for the control of the vehicle with the feature estimation, image based control, and helicopter controller blocks. Various simulations in MATLAB and experiments performed on a model helicopter show that the approach is successful.

As a future work, we plan to experimentally validate the simulation results with stationary and non-stationary objects in the control loop with more advanced motions.

Acknowledgment

This work was partially supported by the Career Project (Project no: 107E211) awarded to Erdinç Altuğ by the Scientific and Technological Research Council of Turkey.

References

[1] F. Chaumette, S. Hutchinson, 'Visual Servo Control Part I: Basic Approaches', 1070-932/06, 2006 IEEE.

[2] C. Lazar, A. Burlacu, 'Dynamic Simulation Model for Image Based Visual Servo Control Systems', IEEE.

[3] E. Marchand, F. Chaumette. 'Feature tracking for visual servoing purposes', Robotics and Autonomous Systems, Vol. 52, No. 1, July2005, pp.:53-70.

[4] E. Altuğ, J. P. Ostrowski, R. Mahony, 'Control of a Quadrotor Helicopter Using Visual Feedback', Proceedings of the 2002 IEEE International Conference on Robotics &Automation Washington, DC 2002.

[5] H. Romero, R. Benosman and R. Lozano. 'Stabilization and Location of a Four Rotor Helicopter Applying Vision', ACC, 3930.3936, Minneapolis, USA, June 2006.

[6] S. Saripalli, J.F. Montgomery and G.S. Sukhatme, 'Visually-guided landing of an UAV', IEEE Trans. on Robotics and Automation, 19(3):371.381, June 2003.

[7] A.D. Wu, E.N. Johnson and A.A. Proctor, 'Vision-Aided Inertial Navigation for Flight Control', AIAA Guidance, Navigation and Control Conference and Exhibition, San Francisco, USA, Aug. 2005.

[8] J.R. Azinheira, P. Rives, J.R.H Carvalho, G.F. Silveira ,E.C. de Paiva and S. S Bueno, 'Visual servo control for the hovering of an outdoor robotic airship', ICRA, 3:2787. 2792, Washington, USA, May 2002.

[9] O. Bourquardez and F. Chaumette, 'Visual servoing of an airplane for auto-landing', IROS, San Diego, USA, Oct. 2007.

[10] A. A. Proctor, E. N. Johnson, T. B. Apker, (2006), 'Vision-only Control and Guidance for Aircraft', Journal of Field Robotics 23(10), 863–890, 2006.

[11] G. S. Sukhatme, L. Mejías, S. Saripalli, P. Campoy, (2005), 'Visual Servoing of an Autonomous Helicopter in Urban Areas Using Feature Tracking', Journal of Field Robotics 23(3/4), 185–199, 2006.

[12] J. Serres, D. Dray, F. Ruffier, N. Franceschini, (2006), 'A Vision-Based Autopilot for a Miniature Air Vehicle: Joint Speed Control and Lateral Obstacle Avoidance', Journal of Autonomous Robot 25: 103–122, 2008.

[13] J. A. Gangloff and M. F. de Mathelin, 'High speed Visual Servoing of a 6 DOF Manipulator using Multivariable Predictive Control', Advances Robotics, vol. 21, no. 10, 2003, pp. 993-1021.

[14] H. Fujimoto, 'Visual Servoing of 6 dof Manipulator by Multirate Control with Depth Identification', Proceedings of 42nd IEEE Conference on Decision and Control, Maui, Hawaii, December 2003, pp. 5408- 5413.

[15] A. Tayebi, S. McGilvray, 'Attitude Stabilization of a VTOL Quadrotor Aircraft'. IEEE Transactions on control systems technology, vol. 14, No.3, 2006.

[16] S. Bouabdallah, P. Murrieri, R. Siegwart, 'Design and Control of an Indoor Micro Quadrotor', Proceedings of the 2004 IEEE conference on Robotics and Automation.

[17] EURON Summer School on Visual Servoing, 'Laboratory Worknotes', Benicàssim, September 16-20, 2002.

[18] Cervera E., "Visual Servoing Toolbox", Jaume I University,Castello,2003, http://sourceforge.net/projects/vstoolbox/

[19] B. Erginer, 'Quadrotor VTOL Aracının Modellenmesi ve Kontrolü' M.S. Thesis (in Turkish), Istanbul Technical University, 2007.

[20] Matrox Imaging Library, http://www.matrox.com/imaging/en/products/software