Proceedings of the 2012 IEEE
International Conference on Robotics and Biomimetics
December 11-14, 2012, Guangzhou, China

Dynamic IBVS Control of an Underactuated UAV

Hamed Jabbari[1,2], Giuseppe Oriolo[2] and Hossein Bolandi[3]

***Abstract*— In this paper image based visual servo approach for 3D translational motion and yaw rotation of an underactuated flying robot is considered. Taking into account the complexity of dynamics of flying robots, main objective of this paper is to consider the dynamics of these robots in designing an image based control strategy. Inertial information of the robot orientation is combined with image information in order to have overall system dynamics in a fashion to apply full dynamic image based controller. Suitable perspective image moments are used in order to have satisfactory trajectories in image space and Cartesian coordinates. A nonlinear controller for the full dynamics of the system is designed. Simulation results are presented to validate the designed controller.**

I. Introduction

Increasing applications of flying robots have motivated researchers to improve their capabilities through developing new control strategies and integrating suitable sensors. Vision sensor has received a great attention for these robots in the last decade and many applications are reported for these robots including estimation of ego-motion [1], pose estimation [2], SLAM [3], automatic landing [4], positioning [5], obstacle avoidance [6] and etc. Some applications, like last three ones just mentioned, need control strategy based on vision information.

There are two major vision based control approaches for robotic systems including image based visual servoing (IBVS), in which control is based on dynamics of image features in image plane, and position based visual servoing (PBVS), where control is on Cartesian space based on 3D information of workspace reconstructed from 2D image data. Vision based control of the flying robots has received a growing interest from late 90's. PBVS approach is reported in [1], [7] and [8]. One of the first works in IBVS control of flying robots is [9] where simplified dynamics model of a blimp beside dynamics of the image features are used to design an image based control low. In [10], it is suggested that for high speed tasks and underactuated robots, dynamics of the robot should be taken into account in vision based control approaches.

For underactuated flying robots, it is not straightforward to design a full dynamic image based visual servo controller. In [5] authors mentioned that dynamics of image features destroy triangular cascade structure of dynamics of flying robots expressed in inertial frame and passivity-like properties of these dynamics can be preserved for centroid image features only when spherical image projection is used. The proposed control law in [5] with a modification in image features is implemented practically on a quadrotor helicopter in [11] and results show undesired behavior in vertical axis. Actually this is the main problem of the spherical moments for image based visual servo control which is addressed and slightly improved by rescaled spherical image moments in [12].

In presented fully dynamic image based visual servo control of flying robot based on spherical image moments [5] [11] [13], inertial orientation information of the robot are used to generate image space error vector except [14] where this term is implicitly considered in an unknown term and this uncertainty is taken into account in stability analysis of the system. Recently an approach based on virtual spring is implemented to design a full dynamic IBVS controller for the quadrotor using perspective image moments [15]. However, the controller for translation along and rotation around vertical axis assumed parallel position of the robot with respect to object and the system is locally stable.

In this paper full dynamic image based visual servo for 3D translation and yaw rotation control of an underactuated flying robot using perspective image moments is considered. The robot is a quadrotor helicopter which recently received a great interest by the researchers. Unlike most of the previous approaches, only roll and pitch angles of the robot are used for vision based control of the quadrotor. Using these information image data are reconstructed on an image plane parallel to the observed object. This reconstruction makes it possible to develop a fully dynamic image based visual servo control of the quadrotor using perspective image moments. A backstepping control approach is used to control the overall system. Simulation results are presented to evaluate the performance of the system.

II. Equations of motion of the robot

In this paper we aim to control an underactuated quadrotor helicopter. This robot has four rotors as its actuators, equipped with four propellers, on a rigid cross frame. Axes of rotation of the propellers are parallel and their air flows point downward. There is a dependency in the lateral motion of the quadrotor where for forward and backward (left and right) motion the robot has to pitch (roll) in the desired direction. This dependency is undesired for IBVS control of these robots where control of translational motion of them is mostly considered by this approach.

[1]H. Jabbari is with Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran. He is also a visiting scholar at the Dipartimento di Informatica e Sistemistica, Universita di Roma "La Sapienza", Rome, Italy. hjabbari at iust.ac.ir

[2]G. Oriolo is with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, Via Ariosto 25, 00185 Rome, Italy. oriolo at dis.uniroma1.it

[3]H. Bolandi is with Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran. hbolandi at iust.ac.ir

978-1-4673-2127-3/12/$31.00

To describe equations of motion of the quadrotor,we consider two coordinate frames. Inertial frame $\mathcal{I} = \{O_i, X_i, Y_i, Z_i\}$ and body fixed frame $\mathcal{B} = \{O_b, X_b, Y_b, Z_b\}$ which is attached to the center of mass of the robot. Center of the frame $\mathcal{B}$ is located in position $\zeta = (x, y, z)$ with respect to the inertial frame and its attitude is given by the orthogonal rotation matrix $R : \mathcal{B} \rightarrow \mathcal{I}$ depending on the three Euler angles ϕ, θ and ψ denoting respectively, the roll, the pitch, and the yaw.

Considering $V \in \Re^3$ and $\Omega = [\Omega_1\ \Omega_2\ \Omega_3]^T \in \Re^3$ respectively as linear and angular velocities of the robot in the body fixed frame, the kinematics of the quadrotor as a 6DOF rigid body will be as follows:

$$\begin{aligned} \dot{\zeta} &= RV \\ \dot{R} &= Rsk(\Omega) \end{aligned} \tag{1}$$

The notation $sk(\Omega)$ is the skew-symmetric matrix such that for any vector $b \in \Re^3$, $sk(\Omega)\, b = \Omega \times b$ where $\times$ denotes the vector cross-product. The relation between time derivative of the Euler angles and the angular velocity Ω is given by,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} \tag{2}$$

Newton-Euler's equations to drive the dynamics of a general 6DOF rigid body, with the mass of m and the constant symmetric inertial matrix $J \in \Re^{3\times3}$ around the center of mass and with respect to the frame $\mathcal{B}$, is as follows:

$$m\dot{V} = -m\Omega \times V + F \tag{3}$$

$$J\dot{\Omega} = -\Omega \times J\Omega + \tau \tag{4}$$

where F and τ are respectively the force and the torque vectors with respect to the frame $\mathcal{B}$ which determine specific dynamics of the system. The quadrotor actuators generate a single actuation thrust input, U_1, and full actuation of the torque $\tau = [U_2\ U_3\ U_4]^T$ which demonstrates underactuated dynamics of the system. The force input F in (3) is as follows

$$F = -U_1 E_3 + mgR^T e_3 \tag{5}$$

where $E_3 = e_3 = [0\ 0\ 1]^T$ are the unit vectors in the body fixed frame and the inertial frame respectively.

III. Image Dynamics

Commonly used image formations for visual servoing are perspective and spherical projection. It is shown in [12] that perspective image moments although have satisfactory practical results for IBVS control of a quadrotor but GAS of the system cannot be guaranteed. In the other hand, although spherical image moments satisfy stability conditions for IBVS but they do not provide suitable behavior in Cartesian coordinates trajectories of the robot. Therefore, in this paper we will use perspective image moments to do IBVS task for the quadrotor. Roll and pitch angles of the robot, and hence camera, are used to reproject perspective image data in a new image plane parallel to the object. Based on selected image features in the new image plane it will be possible to design full dynamic image based controller for these robots.

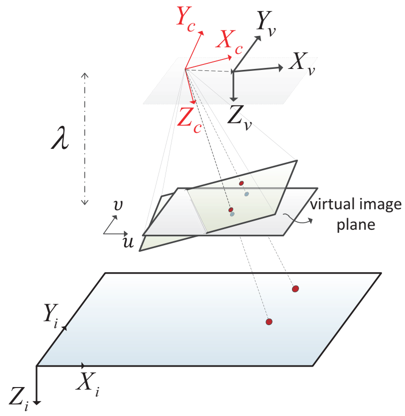

Fig. 1. Camera and virtual coordinate frames and image planes

A. Reprojection of image points

Suppose that ${}^I p$ are the coordinates of a stationary point, P, relative to the inertial frame, $\mathcal{I}$, and ${}^c p(t)$ are the time varying coordinates of P relative to a frame $\mathcal{C} = \{O_c, X_c, Y_c, Z_c\}$, called camera frame, which is attached to the center of projection of a moving camera, where its x and y axes are respectively parallel with horizontal and vertical axes of the image plane and its z axis is perpendicular to the image plane, passes through focal center of the lens and located at a distance of focal lens behind the image plane (Fig. 1). Then we have

$${}^c p(t) = R^T(t)\left[{}^I p - O_c(t)\right]$$

where $R(t)$ is the rotation matrix defining the orientation of the camera frame at time t relative to the inertial frame. We define a new coordinate frame, $\mathcal{V} = \{O_v, X_v, Y_v, Z_v\}$, and image plane, named virtual frame and virtual image plane which have the same alignment to the defined camera frame and the image plane with respect to each other. Virtual frame moves with the camera frame but it does not inherit its dynamics around x and y axis and its center is located with a constant offset, in xy plane of the inertial frame with respect to center of the camera frame, $D = [{}^c x_D\ {}^c y_D\ {}^c z]^T$, (Fig. 1). Thus coordinates of the point P relative to the virtual frame at time t is

$${}^v p(t) = R_\psi^T(t)\left[{}^I p - O_v(t)\right]$$

where R_ψ is the rotation matrix around Z_i. Then time derivative of the ${}^v p(t) = [{}^v x\ {}^v y\ {}^v z]^T$ will be

$$\begin{aligned} \frac{d}{dt}{}^v p(t) &= \left(\frac{d}{dt}R_\psi\right)^T \left({}^I p - O_v\right) - R_\psi^T \dot{O}_v \\ &= -sk\left(\dot{\psi} e_3\right){}^v p - v \end{aligned} \tag{6}$$

For simplicity, time dependency is not shown for variables. In equation (6), vector $\dot{O}_v(t)$ is the linear velocity of the camera frame and also virtual frame with respect to inertial frame and $v(t) = [{}^v v_x\ {}^v v_y\ {}^v v_z]^T$ is the linear velocity of the camera frame expressed in virtual frame.

Considering ${}^{c}p\,(t)$ vector as ${}^{c}p\,(t) = [{}^{c}x\ {}^{c}y\ {}^{c}z]^T$ and well-known perspective projection equations for a camera with focal length of λ, coordinates of the point ${}^{c}p\,(t)$ in the image plane will be

$$u = \lambda \frac{{}^{c}x}{{}^{c}z}, \qquad \nu = \lambda \frac{{}^{c}y}{{}^{c}z}$$

Now utilizing $R_{\phi\theta}$, the rotation matrix respectively around X_i and Y_i, we can reproject the image coordinates (u, ν) to the virtual image plane. We have

$$\begin{bmatrix} {}^{v}u \\ {}^{v}\nu \\ \lambda \end{bmatrix} = \beta R_{\phi\theta} \begin{bmatrix} u \\ \nu \\ \lambda \end{bmatrix} - \begin{bmatrix} q_x^D \\ q_y^D \\ 0 \end{bmatrix}, \tag{7}$$

$$\beta = \lambda / \left([0\ 0\ 1]\, R_{\phi\theta} \begin{bmatrix} u \\ \nu \\ \lambda \end{bmatrix} \right)$$

where $q_x^D = \lambda \frac{{}^{c}x_D}{{}^{c}z}$ and $q_y^D = \lambda \frac{{}^{c}y_D}{{}^{c}z}$ and coefficient β will ensure projection in an image plane with focal length equal to λ.

Remark 1. $({}^{c}x_D, {}^{c}y_D)$ is the distance of the virtual frame from camera frame obtained in desired position of the robot. This displacement is considered to have proper image space error in order to design full dynamic IBVS in section IV. To compute q_x^D and q_y^D in each time, vertical distance and relative yaw rotation of the camera from the observed object are required. This information is available from visual features defined in the remainder of this section. However, in most applications of visual servoing the task is to bring the observed object to the center of image plane where for the defined features in the following we will have $q_x^D = q_y^D = 0$.

In order to obtain the dynamics of the image features in the virtual image plane we will use perspective projection equations of the image features in the new image plane and compute their time derivatives. Using equation (6), the relationship of the velocity of the image coordinates of a point in the virtual plane with the velocity of the camera frame in the matrix form will be

$$\begin{bmatrix} {}^{v}\dot{u} \\ {}^{v}\dot{\nu} \end{bmatrix} = \begin{bmatrix} -\frac{\lambda}{z} & 0 & \frac{{}^{v}u}{z} \\ 0 & -\frac{\lambda}{z} & \frac{{}^{v}\nu}{z} \end{bmatrix} \begin{bmatrix} {}^{v}v_x \\ {}^{v}v_y \\ {}^{v}v_z \end{bmatrix} + \begin{bmatrix} {}^{v}\nu \\ -{}^{v}u \end{bmatrix} \dot{\psi} \tag{8}$$

where $z = {}^{v}z = {}^{c}z$.

B. Image features and their dynamics

In IBVS, to have satisfactory trajectories in both image space and Cartesian coordinates, there should be decoupled and linear link between image space and Cartesian coordinates. In [16], a set of image features based on perspective image moments are presented to image based translational control. In this paper, we use these features for control of the quadrotor. For this purpose, first we define the features and extract their dynamics in the virtual image plane which will be used for image based control. We consider the following assumption for the observed object:

Assumption 1. The observed object is a planar and continuous object and its binary image is obtained by segmentation algorithms.

With considering Assumption 1, image moments m_{ij} of an object consists of n image points are defined as follows [16]

$$m_{ij} = \sum_{k=1}^{n} u_k^i \nu_k^j \tag{9}$$

while centered moments are given by

$$\mu_{ij} = \sum_{k=1}^{n} (u_k - u_g)^i\, (\nu_k - \nu_g)^j \tag{10}$$

where $u_g = \frac{m_{10}}{m_{00}}$ and $\nu_g = \frac{m_{01}}{m_{00}}$. Now we select our image features for translational motion control of the robot as follows

$$q_x = q_z u_g, \quad q_y = q_z \nu_g, \quad q_z = z^* \sqrt{\frac{a^*}{a}} \tag{11}$$

Where z^* is the desired normal distance of the camera from the object and a^* is the desired value of a where we have $a = \mu_{20} + \mu_{02}$.

Using (8) to compute time derivatives of equations (9) and (10) and knowing that $z\sqrt{a} = z^*\sqrt{a^*}$, dynamics of the image features in the new image plane will be

$$\dot{q} = -sk\left(\dot{\psi} e_3\right) \begin{bmatrix} q_x \\ q_y \\ q_z^D \end{bmatrix} - v \tag{12}$$

where q_z^D can be an arbitrary value which will be defined properly to produce image space error in section IV. Equation (12) shows that there is a decoupled and linear link between image space and Cartesian coordinates in translational motion of the robot and also feature dynamics have passivity properties mentioned in [5].

To control the yaw rotation of the robot based on image features we can use object orientation α which is defined by [17]

$$\alpha = \frac{1}{2} \arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \right) \tag{13}$$

Using derivative of equation (10), time derivative of α in new image plane will be

$$\dot{\alpha} = -\dot{\psi} \tag{14}$$

IV. IBVS CONTROLLER

In this section we consider the full dynamic IBVS control of the quadrotor helicopter. The task is to move the camera attached quadrotor to match the observed image features with the predefined desired image features obtained from a stationary object. Before designing the controller we consider the following assumptions:

Assumption 2. The camera frame, C, is coincident with quadrotor body fixed frame, B.

Assumption 3. Image features are always in the field of view of the camera.

Assumption 4. To avoid singularity in equation (2), it is assumed that $-\pi/2 < \theta < \pi/2$.

If Assumption 2 is not satisfied, it is easy to release it just by considering constant transformation between two frames.

Also, based on Assumption 3, Assumption 4 is not restrictive in the image based control.

In order to design the controller, first we need to define error vector in image space. We assume that ${}^{c}q^{d} = \left[q_x^d \; q_y^d \; q_z^d\right]^T$ is the image features of a stationary object obtained in a goal configuration of the robot in the image plane (where $q_x^D = q_x^d$ and $q_y^D = q_y^d$). Then using equation (7), the image error, $q_1 = \left[q_{1x} \; q_{1y} \; q_{1z}\right]^T$, in virtual image plane will be as follows

$$q_1 = q - \left[0 \; 0 \; q_z^d\right]^T \tag{15}$$

where the vector $q = \left[q_x \; q_y \; q_z\right]^T$ is the image features defined in equation (11) and observed in the virtual image plane. Using (12) and assigning $q_z^D = q_z - q_z^d$, derivative of the image error vector will be

$$\dot{q}_1 = -sk\left(\dot{\psi}e_3\right)q_1 - v \tag{16}$$

We write the translational dynamics of the robot, equation (3), in the virtual frame and the attitude dynamics, equation (4), in the body fixed frame. Therefore the remaining dynamics of the system are given by

$$m\dot{v} = -msk\left(\dot{\psi}e_3\right)v + f \tag{17}$$

$$\dot{R}_{\phi\theta} = R_{\phi\theta}sk\left(\Omega\right) - sk\left(\dot{\psi}e_3\right)R_{\phi\theta} \tag{18}$$

$$J\dot{\Omega} = -\Omega \times J\Omega + \tau \tag{19}$$

where equation (18) is obtained using equation (1) and definition of angular velocity. In equation (17) we have

$$f = R_{\phi\theta}F \tag{20}$$

First we will consider the control of the system defined through equations (16)-(19) which leads to translational motion control of the robot and then we will design a controller for the yaw rotation of the robot. For this purpose we will use backstepping approach [18] and follow the method in [5].

At the first step of this approach, our task is to stabilize equation (16) with respect to the following Lyapunov function

$$L_1 = \frac{1}{2}q_1^T q_1$$

The time derivative of it will be

$$\dot{L}_1 = q_1^T\left(-sk\left(\dot{\psi}e_3\right)q_1 - v\right) = -q_1^T v \tag{21}$$

By considering $v = k_1 q_1$ as our control input and substituting in (21) we will have

$$\dot{L}_1 = -k_1 q_1^T q_1$$

If v were available as the control input then equation (16) with $k_1 > 0$ would be stable. Following the standard backstepping approach, we consider an error term q_2 which defines the scaled difference between the considered control input, v, and the actual control input as follows

$$q_2 = \frac{v}{k_1} - q_1 \tag{22}$$

With this definition, time derivative of the q_1 and L_1 will be

$$\dot{q}_1 = -sk\left(\dot{\psi}e_3\right)q_1 - k_1 q_1 - k_1 q_2 \tag{23}$$
$$\dot{L}_1 = -k_1 q_1^T q_1 - k_1 q_1^T q_2$$

Considering equations (17) and (23), derivative of the q_2 will be as follows

$$\dot{q}_2 = \frac{\dot{v}}{k_1} - \dot{q}_1 = -sk\left(\dot{\psi}e_3\right)q_2 + k_1 q_1 + k_1 q_2 + \frac{f}{mk_1} \tag{24}$$

Now we stabilize equation (24) by defining the following Lyapunov function

$$L_2 = L_1 + \frac{1}{2}q_2^T q_2$$

The time derivative of defined Lyapunov function will be

$$\dot{L}_2 = -k_1 q_1^T q_1 + k_1 q_2^T q_2 + q_2^T \frac{f}{mk_1}$$

with $\frac{f}{mk_1}$ as its input, we choose the following controller

$$\frac{f}{mk_1} = -k_2 q_2$$

with $k_2 > k_1$. Similar to the previous step, if the selected control input was available then the equation (24) would be stable. But since the ego-motion dynamics of the camera inherent underactuated dynamics of the quadrotor it is not possible to independently control the position dynamics of the robot through force input and we have to continue the steps of the backstepping approach to reach the actual control inputs of the translation motion of the camera. Thus we consider an error for the defined input and we have

$$q_3 = \frac{f}{mk_1k_2} + q_2 \tag{25}$$

With this definition, derivatives of the q_2 and L_2 will now be

$$\dot{q}_2 = -sk\left(\dot{\psi}e_3\right)q_2 + k_1 q_1 + \left(k_1 - k_2\right)q_2 + k_2 q_3$$
$$\dot{L}_2 = -k_1 q_1^T q_1 + \left(k_1 - k_2\right)q_2^T q_2 + k_2 q_2^T q_3$$

Now we can write the derivative of the q_3 as follows

$$\begin{aligned}\dot{q}_3 = &- sk\left(\dot{\psi}e_3\right)q_3 + k_1 q_1 + \left(k_1 - k_2\right)q_2 + k_2 q_3 \\ &+ \frac{\dot{f}}{mk_1k_2} + sk\left(\dot{\psi}e_3\right)\frac{f}{mk_1k_2}\end{aligned} \tag{26}$$

and define the Lyapunov function L_3 for this dynamics as follows

$$L_3 = L_2 + \frac{1}{2}q_3^T q_3$$

Then the derivative of the L_3 will be

$$\begin{aligned}\dot{L}_3 = &-k_1 q_1^T q_1 + \left(k_1 - k_2\right)q_2^T q_2 \\ &+ q_3^T\left(k_1 q_1 + k_1 q_2 + k_2 q_3 + \frac{\dot{f} + sk\left(\dot{\psi}e_3\right)f}{mk_1k_2}\right)\end{aligned} \tag{27}$$

With $\frac{\dot{f}+sk\left(\dot{\psi}e_3\right)f}{mk_1k_2}$ as its input, we consider the following control

$$\frac{\dot{f} + sk\left(\dot{\psi}e_3\right)f}{mk_1k_2} = -k_3 q_3 \tag{28}$$

By substituting equation (28) in (27) we have

$$\begin{aligned}\dot{L}_3 = & - k_1 q_1^T q_1 + (k_1 - k_2)\, q_2^T q_2 + k_1 q_3^T q_1 + k_1 q_3^T q_2 \\ & + (k_2 - k_3)\, q_3^T q_3 \end{aligned} \tag{29}$$

Using equations (5), (18) and (20) we have

$$\dot{f} + sk\left(\dot{\psi} e_3\right) f = -R_{\phi\theta} sk(\Omega) U_1 E_3 - R_{\phi\theta} \dot{U}_1 E_3 \tag{30}$$

Since equation (30) depends on the angular velocity and torque inputs appear in the dynamics of the derivative of the angular velocity, equation (19), to control orientation of the robot and hence translational motion of it, we continue the procedure with considering a new error term q_4

$$q_4 = \frac{\dot{f} + sk\left(\dot{\psi} e_3\right) f}{m k_1 k_2 k_3} + q_3 \tag{31}$$

Now derivative of the q_3 will be

$$\begin{aligned}\dot{q}_3 = & - sk\left(\dot{\psi} e_3\right) q_3 + k_1 q_1 + (k_1 - k_2)\, q_2 + (k_2 - k_3)\, q_3 \\ & + k_3 q_4\end{aligned}$$

and derivative of the storage function L_3 will be

$$\begin{aligned}\dot{L}_3 = & - k_1 q_1^T q_1 + (k_1 - k_2)\, q_2^T q_2 + k_1 q_3^T q_1 + k_1 q_3^T q_2 \\ & + (k_2 - k_3)\, q_3^T q_3 + k_3 q_3^T q_4 \end{aligned} \tag{32}$$

Now we can write the derivative of the q_4 as follows

$$\begin{aligned}\dot{q}_4 = & \frac{\ddot{f} + sk\left(\ddot{\psi} e_3\right) f + sk\left(\dot{\psi} e_3\right) \dot{f}}{m k_1 k_2 k_3} - sk\left(\dot{\psi} e_3\right) q_3 + k_1 q_1 \\ & + (k_1 - k_2)\, q_2 + (k_2 - k_3)\, q_3 + k_3 q_4 \end{aligned} \tag{33}$$

From equation (18) and (30) we have

$$\begin{aligned} & R_{\phi\theta}^T \left(\ddot{f} + sk\left(\ddot{\psi} e_3\right) f + sk\left(\dot{\psi} e_3\right) \dot{f}\right) + sk\left(\Omega\right) sk\left(\Omega\right) U_1 E_3 \\ & - sk\left(R_{\phi\theta}^T \dot{\psi} e_3\right) sk\left(\Omega\right) U_1 E_3 + 2 sk\left(\Omega\right) \dot{U}_1 E_3 \\ & - sk\left(R_{\phi\theta}^T \dot{\psi} e_3\right) \dot{U}_1 E_3 = \begin{bmatrix} -U_1 \dot{\Omega}_2 \\ U_1 \dot{\Omega}_1 \\ -\ddot{U}_1 \end{bmatrix} \end{aligned} \tag{34}$$

By this equation we can reach actual inputs of the system, $U_1 - U_3$ to control 3D translational motion of the quadrotor. Now we consider the following theorem:

Theorem 1. Under Assumptions 1 to 4, consider the system with dynamics defined by equations (16)-(19) and f and τ as its inputs. These inputs are given through (34) such that

$$\frac{\ddot{f} + sk\left(\ddot{\psi} e_3\right) f + sk\left(\dot{\psi} e_3\right) \dot{f}}{m k_1 k_2 k_3} = sk\left(\dot{\psi} e_3\right) q_3 - k_4 q_4$$

with the following choice of gains

$$k_1 > 0, k_2 > k_1, k_3 > \frac{2k_2^2}{k_2 - k_1}, k_4 > k_3 + 3k_2 \tag{35}$$

Then the errors q_1, q_2, q_3 and q_4 defined respectively by equations (15), (22), (25) and (31) converge asymptotically to zero.

Proof: Consider the following Lyapunov candidate

$$L_4 = L_3 + \frac{1}{2} q_4^T q_4$$

Its derivative using (32) and (33) will be

$$\begin{aligned}\dot{L}_4 = & - k_1 q_1^T q_1 + (k_1 - k_2)\, q_2^T q_2 + k_1 q_3^T q_1 + k_1 q_3^T q_2 \\ & + (k_2 - k_3)\, q_3^T q_3 + k_1 q_4^T q_1 + (k_1 - k_2)\, q_4^T q_2 \\ & + k_2 q_4^T q_3 + (k_3 - k_4)\, q_4^T q_4 \end{aligned} \tag{36}$$

By completing the square for cross terms and using (35) it can be shown that, with the selected control gains, the right-hand side of the equation (36) will be negative definite and this ensures that the errors q_1, q_2, q_3 and q_4 will converge asymptotically to zero. □

The presented control law regulates position and velocity of the camera through regulating error terms q_1 and q_2. Also regulation of the error terms q_3 and q_4 will regulate attitude dynamics of the camera, namely roll and pitch rotation of it.

To control the yaw rotation of the camera we define the following image error

$$q_5 = \alpha - \alpha_d$$

Then according to equation (14) derivative of the error q_5 will be

$$\dot{q}_5 = -\dot{\psi} \tag{37}$$

For simplicity, at this stage we will consider Ω_3 as the input to the system (37). Knowing relation between $\dot{\psi}$ and Ω_3 from equation (2) we consider the following controller

$$\Omega_3 = \left(\frac{\cos\theta}{\cos\phi}\right) \left(k_5 q_5 - \Omega_2 \sin\phi / \cos\theta\right) \tag{38}$$

By substituting equation (38) in (37) we will have following dynamics for the error q_5

$$\dot{q}_5 = -k_5 q_5$$

where with $k_5 > 0$ the error term q_5 will exponentially converge to zero.

V. Simulation results

In this section we show MATLAB® simulations to evaluate the performance of the proposed visual servo controller. The camera frame rate is 50 Hz and sampling time for the rest of the system is 1 ms. In the simulations the robot is assumed to be in a hover position with desired object in the camera field of view. The visual information include coordinates of four points related to the four vertexes of a rectangle which are used to calculate image features defined in (11) and (13). Vertexes of the rectangle with respect to the inertial frame are located at $(0.25, 0.2, 0)$, $(0.25, -0.2, 0)$, $(-0.25, 0.2, 0)$, $(-0.25, -0.2, 0)$. These points are projected with perspective projection on a digital image plane with focal length divided by pixel size (identical in both u and v directions) equal to 213 and principal point located at (80,60). Parameters used for the dynamics model of the robot, equations (3) and (4), are $m = 2$, $g = 9.81$ and $J = diag\,(0.0081, 0.0081, 0.0142)$. Proportional controller with unit gain is used to control the angular velocity Ω_3.

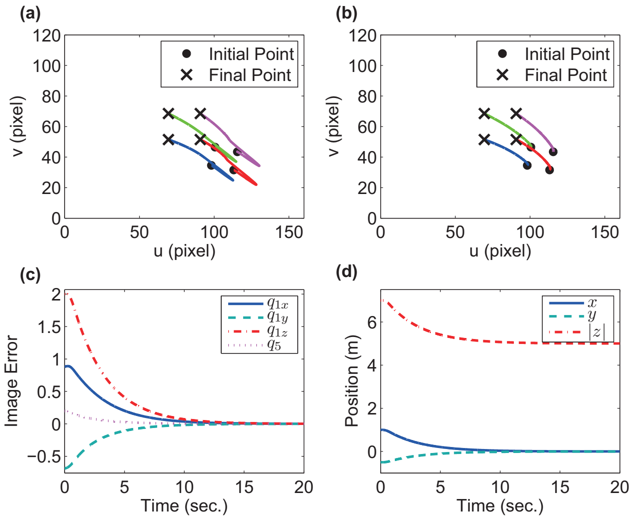

Fig. 2. Simulation 1: (a) point features trajectories in the image plane, (b) point features trajectories in the virtual image plane, (c) image space errors and (d) time evolution of the UAV Cartesian coordinates.

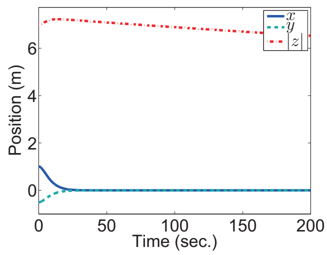

Fig. 3. Time evolution of the UAV Cartesian coordinates for simulation 1 using IBVS controller of [5].

In the first simulation, the robot is assumed in hover condition and its initial position is at $(1, -0.5, -7)$ with respect to the inertial frame and the desired image features are obtained at $(0, 0, -5)$. The value of the image feature α at these positions are 0.2 and 0 respectively. The control gains are chosen to be $k_1 = 0.3$, $k_2 = 2.4$, $k_3 = 6$, $k_4 = 10$ and $k_5 = 0.4$. Fig. 2 shows the results for the first simulation. As expected, results show satisfactory trajectories in both the Cartesian coordinates and the virtual image plane.

As a comparison, in the second simulation, spherical image moments, used in [5] and [14], are utilized based on the IBVS controller designed in [5]. Initial and final positions of the quadrotor are as same as the first simulation. Fig. 3 shows the Cartesian coordinates trajectories of the robot and undesired behavior in z direction is obvious in the result.

VI. Conclusion

In this paper, a fully dynamic image based visual servoing approach has been developed for controlling translational motion and yaw rotation of a quadrotor equipped with an on-board camera. Inertial orientation information of the robot in combination with image data are used to deal with problem of dynamic image based visual servo control of these robots. Suitable perspective image moments are used to have satisfactory trajectories in both image space and Cartesian coordinates. Simulations results show satisfactory response of the presented visual servo approach for positioning task of the robot observing a stationary object.

However, for the selected image features, prior information for the observed object is required, where it is not always available in practice. Also just a simple dynamic model of the robot with known parameters is considered. Thus, our future work is to design an image based controller dealing with the possible uncertainties in both image features and dynamic model of the robot and also improve the conditions to keep the visual features in the field of view of the camera.

References

[1] O. Shakernia, Y. Ma, T. J. Koo, T. John, and S. Sastry, "Landing an unmanned air vehicle: Vision based motion estimation and nonlinear control," *Asian Journal of Control*, vol. 1, pp. 128–145, 1999.

[2] E. Altug, J. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback," in *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, vol. 1, 2002, pp. 72 – 77 vol.1.

[3] F. Caballero, L. Merino, J. Ferruz, and A. Ollero, "Vision-based odometry and slam for medium and high altitude flying uavs," *Journal of Intelligent Robotic Systems*, vol. 54, pp. 137–161, 2009.

[4] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme, "Visually-guided landing of an unmanned aerial vehicle," *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 371–381, 2003.

[5] T. Hamel and R. Mahony, "Visual servoing of an under-actuated dynamic rigid-body system: an image-based approach," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 2, pp. 187 –198, apr 2002.

[6] A. Beyeler, J.-C. Zufferey, and D. Floreano, "Vision-based control of near-obstacle flight," *Autonomous Robots*, vol. 27, pp. 201–219.

[7] J. P. Ostrowski and C. J. Taylor, "Control of a quadrotor helicopter using dual camera visual feedback," *International Journal of Robotics Research*, p. 2005.

[8] L. Garca Carrillo, E. Rondon, A. Sanchez, A. Dzul, and R. Lozano, "Stabilization and trajectory tracking of a quad-rotor using vision," *Journal of Intelligent Robotic Systems*, vol. 61, pp. 103–118, 2011.

[9] H. Zhang and J. Ostrowski, "Visual servoing with dynamics: control of an unmanned blimp," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 1, 1999, pp. 618 –623 vol.1.

[10] F. Chaumette and S. Hutchinson, "Visual servo control. ii. advanced approaches [tutorial]," *Robotics Automation Magazine, IEEE*, vol. 14, no. 1, pp. 109 –118, march 2007.

[11] N. Guenard, T. Hamel, and R. Mahony, "A practical visual servo control for an unmanned aerial vehicle," *Robotics, IEEE Transactions on*, vol. 24, no. 2, pp. 331 –340, april 2008.

[12] 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," *Robotics, IEEE Transactions on*, vol. 25, no. 3, pp. 743 –749, June 2009.

[13] F. Le Bras, T. Hamel, R. Mahony, and A. Treil, "Output feedback observation and control for visual servoing of vtol uavs," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 9, pp. 1008–1030, 2011.

[14] T. Hamel and R. Mahony, "Image based visual servo control for a class of aerial robotic systems," *Automatica*, vol. 43, no. 11, pp. 1975–1983, 2007.

[15] 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*, may 2011, pp. 5670 –5676.

[16] O. Tahri and F. Chaumette, "Point-based and region-based image moments for visual servoing of planar objects," *Robotics, IEEE Transactions on*, vol. 21, no. 6, pp. 1116 – 1127, dec. 2005.

[17] F. Chaumette, "Image moments: a general and useful set of features for visual servoing," *Robotics, IEEE Transactions on*, vol. 20, no. 4, pp. 713 – 723, aug. 2004.

[18] I. K. M. Krstic and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*. John Wiley & Sons, 1995.