Image-Based Visual Servo Control of an Underactuated Flying Robot Tracking a Moving Target

Hamed Jabbari Asl\textsuperscript{1} and Hossein Bolandi\textsuperscript{2}

\textbf{Abstract}—In this paper, dynamic image-based visual servo control approach is designed for an underactuated flying robot tracking a moving target. The objective is to consider the full dynamics of the system to design a vision based controller. Passivity properties of dynamics of perspective image moments in virtual image plane, obtained using inertial information of the robot, are utilized to design a dynamic image based visual servo controller. A nonlinear robust controller based on backstepping approach is designed to deal with uncertainties in dynamics of image features related to depth information and motion of the target. The controller is also robust with respect to uncertainties in dynamics of the robot. Simulation results are presented to validate the designed controller.

\textit{Keywords: Visual servoing, quadrotor, robust controller}

I. INTRODUCTION

Vision based control of robots mainly includes two major approaches 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.

Attention to control flying robots using vision based methods raised from the late 90’s. In [1], PBVS approach is used to control a helicopter by estimating pose and ego-motion of it using a camera. A special camera configuration is utilized in [2] to estimate pose of a quadrotor helicopter and based on these estimations two different nonlinear controllers for the dynamics of the robot are developed. Position based approach is also implemented in [3] to land a quadrotor helicopter on the ground.

Image based approach is reported in [4] for the quadrotor with an adaptive method to keep the image features in the field of view of the camera. In [5], IBVS is used to land the quadrotor helicopter on a moving platform. However, in these works, the controller for the dynamics of the robot is separately designed with respect to the dynamics of image features and uncertainties of the dynamics of the system related to depth information of image features and motion of the target are not considered.

The importance of designing a full dynamic vision based controller for high speed tasks and underactuated robots is reported in many works including [6]. In [7] authors have designed a cascade nonlinear controller for IBVS control of an underactuated quadrotor helicopter. The proposed control law with a modification in image features is implemented practically on an underactuated quadrotor helicopter in [8] and result shows undesired behavior in vertical axis. This problem of the spherical moments for image based visual servo control is addressed and slightly improved by rescaled spherical image moments in [9]. To deal with the mentioned problem, an approach based on virtual spring philosophy is proposed in [10] to design a full dynamic IBVS controller for quadrotor using perspective image moments. However, the controllers for translation along and rotation around vertical axis have assumed parallel position of the robot with respect to object and the system is locally stable.

Beside the mentioned problem, there are some sources of disturbances that greatly affect performance of IBVS approach for flying robot and can drive the system to be unstable. Disturbances include both in dynamics of image features, when the target is not stationary, and also dynamics of robot.

In [11], we showed that dynamics of some image features obtained from perspective image moments have passivity-like properties in a oriented image plane, called virtual image plane. Using this property it is possible to design a controller for the cascade dynamics of the system for IBVS control of underactuated quadrotor helicopter. In this paper, we use the same idea to design a robust IBVS controller for a quadrotor flying on a moving target. The main objective of this paper is to consider uncertainties of the image dynamics, related to depth information and motion of the target, and also uncertainties in the dynamics of the robot in designing IBVS control approach.

II. EQUATIONS OF MOTION OF THE ROBOT

In this section we describe kinematic and dynamic models of quadrotor helicopter. The models are similar to those introduced in the literature including [12] and [13]. To describe equations of motion of the quadrotor, equipped with a down-looking camera, which define the ego-motion of the camera, 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 $\phi$, $\theta$, and $\psi$ denoting, respectively, the roll, the pitch, and the yaw.

Considering $V \in \mathbb{R}^3$ and $\Omega = [\Omega_1, \Omega_2, \Omega_3]^T \in \mathbb{R}^3$ 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

\[
\dot{\mathbf{q}} = \mathbf{Rv}
\]
\[
\dot{\mathbf{R}} = \mathbf{Rsk}(\Omega)
\]

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

\[
\begin{bmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\
0 & \cos \phi & -\sin \phi \\
0 & \sin \phi \sec \theta & \cos \phi \sec \theta
\end{bmatrix}
\begin{bmatrix}
\Omega_1 \\
\Omega_2 \\
\Omega_3
\end{bmatrix}
\]

\[
(2)
\]

Newton-Euler’s equations to derive the dynamics of a general 6DOF rigid body, with the mass of \(m\) and the constant symmetric inertial matrix \(\mathbf{J} \in \mathbb{R}^{3 \times 3}\) around the center of mass and with respect to the frame \(B\), is as follows

\[
\ddot{\mathbf{X}} = -\mathbf{\Omega} \times \mathbf{V} + \mathbf{F} + \mathbf{\Delta}_2
\]
\[
\mathbf{J}\mathbf{\Omega} = -\mathbf{\Omega} \times \mathbf{J}\mathbf{\Omega} + \mathbf{\tau} + \mathbf{\Delta}_3
\]

\[
(3)
\]

\[
(4)
\]

where \(\mathbf{F} \in \mathbb{R}^3\) and \(\mathbf{\tau} \in \mathbb{R}^3\) are respectively the force and the torque vectors with respect to the frame \(B\) which determine specific dynamics of the system and \(\mathbf{\Delta}_2 \in \mathbb{R}^3\) and \(\mathbf{\Delta}_3 \in \mathbb{R}^3\) are the unstructured forces and moments to the translational and rotational dynamics of the system respectively. The inertial matrix is diagonal, \(\mathbf{J} = \text{diag}(J_{xx}, J_{yy}, J_{zz})\), where \(O_b\) is coincide with the body principal axis of inertial. For the quadrotor helicopter, main contributions of \(\mathbf{F}\) and \(\mathbf{\tau}\) include forces and torques generated by actuators of the robot and the gravitational force due to the gravity \(g\). The quadrotor actuators generate a single actuation trust input, \(U_1\), and full actuation of the torque \(\mathbf{\tau} = [U_2\ U_3\ U_4]^T\) which demonstrates underactuated dynamics of the system. The force input \(\mathbf{F}\) in (3) is as follows

\[
\mathbf{F} = -\frac{1}{m}U_1\mathbf{E}_3 + g\mathbf{R}^T\mathbf{e}_3
\]

\[
(5)
\]

where \(\mathbf{E}_3 = \mathbf{e}_3 = [0\ 0\ 1]^T\) are the unit vectors in the body fixed frame and the inertial frame respectively.

### III. IMAGE DYNAMICS

Perspective and spherical projections are two projection methods that commonly used for vision based control of robots. Image features obtained from perspective projection although have satisfactory properties for IBVS, but it is not possible to guarantee stability of the system when they are utilized for underactuated robots [9]. In order to solve this problem, in [11] we proposed a reprojection method using roll and pitch angles of the robot which makes it possible to design full dynamic image based controller for the quadrotor by selecting a suitable set of image features.

By defining a virtual frame, \(\mathcal{V} = \{O_v, X_v, Y_v, Z_v\}\), as in [11], coordinates of a moving point \(P\) relative to the virtual frame at time \(t\) will be

\[
{p}(t) = \mathbf{R}_\psi^T(t) [p(t) - O_v(t)]
\]

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

\[
\frac{d}{dt} \mathbf{v}(t) = \left( \frac{d}{dt} \mathbf{R}_\psi \right)^T (I - \mathbf{O}_v) - \mathbf{R}_\psi^T \mathbf{O}_v + \mathbf{R}_\psi^T \mathbf{p}
\]

\[
= -sk(\psi\mathbf{e}_3)v - \mathbf{P} - \mathbf{\Delta}_1
\]

\[
(6)
\]

In (6), vector \(\mathbf{v}(t) = [v_x\ v_y\ v_z]^T\) is the linear velocity of the camera frame expressed in the virtual frame and \(\mathbf{\Delta}_1(t) = [d_x\ d_y\ d_z]^T\) is the velocity of the moving point in the virtual frame.

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

\[
\begin{align*}
\mathbf{u} &= \lambda \frac{x}{c_x} \\
\mathbf{v} &= \lambda \frac{y}{c_y}
\end{align*}
\]

Now utilizing \(\mathbf{R}_{ps}\), the rotation matrix respectively around \(X_i\) and \(Y_i\), we can reproject the image coordinates \((u, \nu)\) to the virtual image plane. We have [11]

\[
\begin{bmatrix}
\lambda \mathbf{u} \\
\lambda \mathbf{v}
\end{bmatrix} = \mathbf{R}_{ps}(\mathbf{u})
\]

\[
(7)
\]

where \(q_{ps}^D = \lambda \frac{x}{c_x} \). Using (6), relationship of the velocity of the image coordinates of a moving point in the virtual plane with the velocity of the camera frame in a matrix form will be

\[
\begin{bmatrix}
\mathbf{v}_x \\
\mathbf{v}_y \\
\mathbf{v}_z
\end{bmatrix} = \left[ \begin{array}{ccc}
\lambda & 0 & \frac{x}{c_x} \\
0 & \lambda & \frac{y}{c_y} \\
\frac{-x}{c_x} & \frac{-y}{c_y} & \lambda
\end{array} \right] \left[ \begin{array}{c}
\mathbf{v}_x \\
\mathbf{v}_y \\
\mathbf{v}_z
\end{array} \right] + \left[ \begin{array}{c}
\mathbf{v}_u \\
\mathbf{v}_u \\
\mathbf{v}_u
\end{array} \right]
\]

\[
(8)
\]

where \(z = \mathbf{v}_z\).

Now by using image moments \(m_{ij}\) and centered image moments \(\mu_{ij}\), we define image features for translational motions control of the robot as follows

\[
q_x = q_x \frac{u_x}{\alpha} \quad q_y = q_y \frac{u_y}{\alpha} \quad q_z = \sqrt{\frac{a^2}{\alpha}}
\]

\[
(9)
\]

where \(u_x = m_{10}/m_{00}, \ u_y = m_{01}/m_{00}\) and \(a^*\) is the desired value of \(a\) which \(\alpha\) is defined as follows

\[
\alpha = v^2 \mu_{20} + v^2 \mu_{02}
\]

By using (7) and knowing that \(z = \sqrt{a^*} = \sqrt{\lambda \mathbf{v}_z}\), where \(z^*\) is the desired normal distance of the camera from the object, dynamics of the features in the new image plane will be

\[
\dot{q} = -sk(\psi\mathbf{e}_3) \left[ \begin{array}{c}
q_x \\
q_y \\
q_z
\end{array} \right] = \frac{1}{z^*}v + \frac{1}{z^*} \mathbf{\Delta}_1
\]

\[
(10)
\]

where \(q = [q_x\ q_y\ q_z]^T\) is the vector of image features defined in (8) and observed in virtual image plane and \(q_{ps}^D\) can be an arbitrary value which will be defined properly to produce image space error in section IV.
IV. ROBUST IBVS CONTROLLER

In this section we consider the full dynamic IBVS control of the quadrotor helicopter which will be robust with respect to uncertainty in the dynamics of image features and uncertainty in dynamics of the robot. Task is to move the camera attached quadrotor to match the observed image features with the predefined desired image features obtained from an object. The object is considered to have 3D translation motion and yaw rotation, which can be a wheeled mobile robot moving on a flat ground or a quadrotor flying with small variations in the roll and the pitch angles. These kinds of motion for the object will satisfy our assumption to have a planar image target. The visual servo approach is only used to control the translation motion of the quadrotor. It is also possible to control the yaw rotation of the robot through visual information as done in [11] but here we assume that the yaw rotation is controlled by IMU data to have stable velocity.

Before designing the controller, it is assumed that the camera frame, C, is coincident with the quadrotor body fixed frame, B. If this condition does not hold, we just need a constant transformation between the two frames.

In order to preserve the passivity properties of image dynamics, we will consider the case that our desired image features are as follows

\[
q^d = \begin{bmatrix} d_x & d_y & d_z \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & q^d_z \end{bmatrix}^T
\]

It is common in visual servoing of quadrotor to have the observed object in the center of the image plane. This task for our selected image features, (8), will lead to \(q^d = q^d_{yz} = 0\). Now we define the image error for translational motions control of the robot as follows

\[
q_1 = q - \begin{bmatrix} 0 & 0 & q_z^d \end{bmatrix}^T
\]

Using (9) and assigning \(q^d_{yz} = q_z - q^d_z\), the derivative of the image error vector will be

\[
q_1 = -sk\left(\psi e_3\right)q_1 - \frac{1}{z^*}v + \frac{1}{z^*}\Delta_1
\]

Now we can write the full dynamics of the image features in the virtual plane. To do this, we write the translational dynamics of the robot, (3), in the virtual frame and the attitude dynamics, (4), in the body fixed frame. Then we will have the full dynamics of the system given by

\[
\dot{q}_1 = -sk\left(\psi e_3\right)q_1 - \delta v + \delta \Delta_1
\]

\[
\dot{v} = -sk\left(\psi e_3\right)v + f + \Delta_2
\]

\[
\dot{R}_{\phi\theta} = R_{\phi\theta}sk\left(\Omega\right) - sk\left(\psi e_3\right)R_{\phi\theta}
\]

\[
J\dot{\Omega} = -\Omega \times J\Omega + \tau + \Delta_3
\]

where (13) is obtained using (1) and definition of the angular velocity. In (11), \(\delta = \frac{1}{z^*}\) and in (12) we have

\[
f = R_{\phi\theta}F
\]

Since velocity of the motion of the target is not always available, hence, \(\Delta_1\) in (11) is considered as a disturbance to the system. Beside this uncertainty, the \(\delta\) is an unknown parameter. A robust controller is designed to deal with these uncertainties and also uncertainties in dynamics of translational and rotational motions of the robot. Before developing the controller, we consider the following assumption

**Assumption 1.** The positive parameter \(\delta\) and uncertainty \(\Delta_1\) in (11) and also uncertainties \(\Delta_2\) and \(\Delta_3\) respectively in (12) and (14) are assumed to be upper bounded such that

\[
\delta \leq \delta_{\text{max}}, \quad \|\Delta_1\| \leq D_1, \quad \|\Delta_2\| \leq D_2 \quad \|\Delta_3\| \leq D_3
\]

where \(\delta_{\text{max}}, D_1, D_2\) and \(D_3\) are known (in this paper, \(\|x\|\) is Euclidean norm of the vector \(x\)).

Now we define the following error signals

\[
q_2 = \frac{v}{k_1} - q_1
\]

\[
q_3 = \frac{f}{k_1k_2} + q_2
\]

\[
q_4 = \frac{sk\left(\psi e_3\right)f}{k_1k_2k_3} + q_3
\]

With considering the following scalar values

\[
h_1 = \frac{D_3}{mk_1k_2k_3J_{yy}}
\]

\[
h_2 = \frac{D_3}{mk_1k_2k_3J_{xx}}
\]

where \(k_1, k_2\) and \(k_3\) are constant gains and the following vector

\[
A_p = sk\left(\Omega\right)sk\left(\Omega\right)U_1E_3 - sk\left(R_{\phi\theta}^T\psi e_3\right)sk\left(\Omega\right)U_1E_3
\]

\[
+ 2sk\left(\Omega\right)U_1E_3 - sk\left(R_{\phi\theta}^T\psi e_3\right)U_1E_3
\]

and also defining \([x]_i\), as the \(i\)’s element of the vector \(x\), we present the following theorem for image based translational motions control of the quadrotor helicopter

**Theorem 1.** Consider the system dynamics (11)-(14) with inputs \(U_1, U_2\) and \(U_3\) under assumption 1. We propose the following IBVS controller

\[
- U_1 \begin{bmatrix} \frac{J_{yy}}{J_{yy}} & \frac{J_{yx}}{J_{yy}} & \frac{J_{yz}}{J_{yy}} \end{bmatrix} + \frac{A_p}{J_{yy}}
\]

\[
- U_1 \begin{bmatrix} -J_{xx} - J_{yx}\Omega_2\Omega_3 \end{bmatrix} + A_p
\]

\[
+ mk_1k_2k_3R_{\phi\theta}^T \begin{bmatrix} sk\left(\psi e_3\right)q_3 + k_2q_2 - k_3q_3 - k_4q_4 \end{bmatrix}
\]
where the control gains satisfy the following conditions

\[
\begin{align*}
    k_1 & > \frac{\delta_{\text{max}}}{2}, \\
    k_2 & > \delta_{\text{max}} k_1 + \frac{\delta_{\text{max}}^2}{2} + \frac{1}{2}, \\
    k_3 & > \frac{\delta_{\text{max}}^2}{2} + \frac{1}{2} + \frac{\delta_{\text{max}} k_2^2}{k_1 - \frac{\delta_{\text{max}}}{2}}, \\
    k_4 & > \frac{\delta_{\text{max}}^2}{2} + \frac{1}{2} + \frac{\delta_{\text{max}} k_2^2}{2 k_1 - \frac{\delta_{\text{max}}}{2}} \\
    & + \frac{\delta_{\text{max}}^2}{2} k_1^2 - \delta_{\text{max}} \delta_{\text{max}} - 1 
\end{align*}
\]  

(20)

and \( \epsilon_i(t) \), \( i = 1, 2 \) are positive bounded variables and \( \kappa \) is equal to 0.2785. Then the error signals \( q_1, q_2, q_3, q_4 \), which are defined respectively in (10), (16), (17) and (18), are uniformly ultimately bounded and converge exponentially to a small ball containing the equilibrium point of the system, \( q_1 = q_2 = q_3 = q_4 = 0 \).

**Proof:** The controller is designed using backstepping approach [14]. So, we will show steps of design by considering (11) as the first dynamics that should be controlled. Then we select the following Lyapunov function to stabilize this dynamics

\[
L_1 = \frac{1}{2} q_1^T q_1
\]

Using (11), derivative of this function will be

\[
\dot{L}_1 = q_1^T \left( -s k (\dot{\psi}_e) q_1 - \delta v + \delta \Delta_1 \right) = -s q_1^T v + q_1^T \delta \Delta_1
\]

(21)

by selecting \( v = k_1 q_1 \) as virtual control input to the dynamics (11) and substituting in (21) we will have

\[
\dot{L}_1 = -\delta k_1 q_1^T q_1 + q_1^T \delta \Delta_1
\]

Since \( v \) is not the actual input, the error signal \( q_2 \) is defined as in (16). With this definition, dynamics (11) and the derivative of \( L_1 \) will be as follows

\[
\dot{q}_1 = -s k (\dot{\psi}_e) q_1 - \delta k_1 q_1 - \delta k_2 q_2 + \delta \Delta_1
\]

\[
\dot{L}_1 = -\delta k_1 q_1^T q_1 - \delta k_1 q_1^T q_2 + q_1^T \delta \Delta_1
\]

(22)

Using (12) and (22) dynamics of vector \( q_2 \) will be

\[
\dot{q}_2 = -s k (\dot{\psi}_e) q_2 + \delta k_1 q_1 + \delta k_2 q_2 + \frac{f}{mk_1} + \frac{\Delta_2}{mk_1} - \delta \Delta_1
\]

(23)

Now we define the following Lyapunov function for the dynamics of \( q_2 \)

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

By using (22) and (23) the derivative of this function will be

\[
\dot{L}_2 = -\delta k_1 q_1^T q_1 + \delta k_1 q_1^T q_2 + \frac{f}{mk_1} + (q_1 - q_2)^T \delta \Delta_1 + \frac{\Delta_2}{mk_1} q_2^T q_2
\]

(24)

At this step, we select \( \frac{f}{mk_1} = -k_1 q_1 \) as the control input to dynamics (24). Since the dynamics of the camera inherit the underactuated dynamics of the quadrotor, it is not possible to independently control the position dynamics of it through force input and so we continue steps of the backstepping approach by defining the error vector \( q_3 \) as in (17) to reach the actual control inputs of the translation motion of the camera. Then, we can write (23) and (24) as follows

\[
\dot{q}_2 = -s k (\dot{\psi}_e) q_2 + \delta k_1 q_1 + \delta k_2 q_2 + k_2 q_3 - \delta \Delta_1 + \frac{\Delta_2}{mk_1}
\]

(25)

Now from (24), the dynamics of \( q_3 \) will be

\[
\dot{q}_3 = -s k (\dot{\psi}_e) q_3 + \delta k_1 q_1 + \delta k_2 q_2 + k_2 q_3 + \frac{\dot{f}}{mk_1 k_2} + s k (\dot{\psi}_e) q_3 + \frac{f}{mk_1 k_2} - \delta \Delta_1 + \frac{\Delta_2}{mk_1}
\]

(26)

The following Lyapunov function is considered for this dynamics

\[
L_3 = \frac{1}{2} q_3^T q_3
\]

Using (25) and (26), derivative of this function will be

\[
\dot{L}_3 = -\delta k_1 q_1^T q_1 - \delta k_1 q_1^T q_2 + k_2 q_3^T q_3
\]

\[
+ \delta k_1 q_1^T q_1 + \delta k_1 q_3^T q_2 + (q_2 + q_3)^T \frac{\Delta_2}{k_1}
\]

\[
+ (q_1 - q_2^T q_3^T \delta \Delta_1 + q_3^T \frac{\dot{f} + s k (\dot{\psi}_e) f}{mk_1 k_2}
\]

(27)

We consider \( \frac{\dot{f} + s k (\dot{\psi}_e) f}{mk_1 k_2} = -k_3 q_3 \) as the input to the dynamics (26). From (5), (13) and (15) we have

\[
\dot{f} + s k (\dot{\psi}_e) f = -R_{\theta \phi} sk(\Omega) U_1 E_3 - R_{\theta \phi} U_1 E_3
\]

(28)

Since (28) depends on the angular velocities and the torque inputs, to control orientation and hence translational motions of the robot, appear in the dynamics of the derivative of angular velocities, equation (14), therefore we continue the backstepping procedure with defining \( q_4 \) as in (18) to reach the actual inputs of the system. Then (26) and (27) will be as follows

\[
\dot{q}_3 = -s k (\dot{\psi}_e) q_3 + \delta k_1 q_1 + \delta k_1 q_2 + k_2 q_3
\]

\[
+ (q_2 + q_3)^T \frac{\Delta_2}{mk_1}
\]

\[
L_3 = -\delta k_1 q_1^T q_1 - \delta k_1 q_1^T q_2 + k_2 q_3^T q_3
\]

\[
+ \delta k_1 q_1^T q_1 + \delta k_1 q_3^T q_2 + k_3 q_4^T q_4 + (q_2 + q_3)^T \frac{\Delta_2}{mk_1}
\]

(30)
From (13) and (28) we have
\[
\mathbf{R}_{\theta}^{\top} \left[ \ddot{\mathbf{f}} + s k \left( \ddot{\mathbf{e}}_3 \right) + s k \left( \ddot{\mathbf{e}}_3 \right) \right] + \mathbf{A}_p = \begin{bmatrix}
-U_1 \Omega_2 \\
U_1 \Omega_1 \\
-U_1
\end{bmatrix}
\tag{31}
\]
Now using (14), (29) and (31) we can write dynamics of \( \mathbf{q}_4 \) as follows
\[
\dot{\mathbf{q}}_4 = - s k \left( \ddot{\mathbf{e}}_3 \right) \mathbf{q}_3 + \delta k_1 \mathbf{q}_1 + \left( \delta k_1 - k_2 \right) \mathbf{q}_2 \\
+ \left( k_2 - k_3 \right) \mathbf{q}_3 + k_3 \mathbf{q}_4 - \Delta_1 \frac{\Delta_1}{k_1} - \frac{\mathbf{R}_{\theta} \mathbf{A}_p}{mk_1 k_2 k_3} \\
+ \frac{\mathbf{R}_{\theta} U_1}{mk_1 k_2 k_3} \begin{bmatrix}
- \frac{\Delta_1}{k_2} \\
\frac{\Delta_1}{k_3} \\
0
\end{bmatrix}
\tag{32}
\]
Finally we consider the following Lyapunov function
\[
L_4 = L_3 + \frac{1}{2} \mathbf{q}_4 \mathbf{q}_4 ^{\top}
\tag{33}
\]
Then using (30), (32) and the controller defined in (19), the derivative of Lyapunov function \( L_4 \) will be
\[
\begin{align*}
\dot{L}_4 &= -\delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_1 - \left( k_2 - k_3 \right) \mathbf{q}_2 ^{\top} \mathbf{q}_2 - \left( k_3 - k_2 \right) \mathbf{q}_3 ^{\top} \mathbf{q}_3 \\
&- \left( k_3 - k_4 \right) \mathbf{q}_4 ^{\top} \mathbf{q}_4 + \delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_1 + k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_2 \\
&+ \delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_2 + \delta \left\| \mathbf{q}_1 \right\| \left\| \Delta_1 \right\| + \delta \left\| \mathbf{q}_2 \right\| \left\| \Delta_1 \right\| + \delta \left\| \mathbf{q}_3 \right\| \left\| \Delta_1 \right\| \\
&+ \delta \left\| \mathbf{q}_4 \right\| \left\| \Delta_1 \right\| + \left\| \Delta_2 \right\| \frac{\Delta_2}{mk_1} \\
&+ \left\| \mathbf{q}_3 \right\| \left\| \Delta_2 \right\| \frac{\Delta_2}{mk_1} \\
&+ \mathbf{q}_1 ^{\top} \mathbf{R}_{\theta} \mathbf{U}_1 \begin{bmatrix}
\frac{\left\| \Delta_1 \right\|}{mk_1} \\
\frac{\left\| \Delta_1 \right\|}{mk_1} \\
0
\end{bmatrix}
\end{align*}
\tag{34}
\]
Since \( \delta \) is positive and knowing the fact that for any vector \( \mathbf{x} \in \mathbb{R}^3 \) we have \( \left\| \mathbf{Rx} \right\| = \left\| \mathbf{x} \right\| \), then we can write
\[
\dot{L}_4 \leq -\delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_1 - \left( k_2 - k_3 \right) \mathbf{q}_2 ^{\top} \mathbf{q}_2 - \left( k_3 - k_2 \right) \mathbf{q}_3 ^{\top} \mathbf{q}_3 \\
- \left( k_3 - k_4 \right) \mathbf{q}_4 ^{\top} \mathbf{q}_4 + \delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_1 + k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_2 \\
+ \delta k_1 \mathbf{q}_1 ^{\top} \mathbf{q}_2 + \delta \left\| \mathbf{q}_1 \right\| \left\| \Delta_1 \right\| + \delta \left\| \mathbf{q}_2 \right\| \left\| \Delta_1 \right\| + \delta \left\| \mathbf{q}_3 \right\| \left\| \Delta_1 \right\| \\
+ \delta \left\| \mathbf{q}_4 \right\| \left\| \Delta_1 \right\| + \left\| \mathbf{q}_2 \right\| \frac{\Delta_2}{mk_1} \\
+ \left\| \mathbf{q}_3 \right\| \frac{\Delta_2}{mk_1} \\
+ \mathbf{q}_1 ^{\top} \mathbf{R}_{\theta} \mathbf{U}_1 \begin{bmatrix}
\frac{\left\| \Delta_1 \right\|}{mk_1} \\
\frac{\left\| \Delta_1 \right\|}{mk_1} \\
0
\end{bmatrix}
\]
V. SIMULATION RESULTS

In this section we present MATLAB® simulations to evaluate the performance of the proposed robust visual servo controller. In the simulations, the camera frame rate is 50 Hz and sampling time for the rest of the system is 1 ms. The robot is assumed to start in a hover position with the desired object in the camera field of view. Visual information include coordinates of four point related to the four vertexes of a rectangle which are used to calculate image features defined in (8). 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 pointes are projected with perspective projection on a digital image plane with focal length divided by pixel size (identical in both u and v directions) equals to 213 and the principle point is located at (80,60).

Parameters used for the dynamics model of the robot, (3) and (4), are \( m = 2 \text{kg}, \ g = 9.81 \text{m/s}^2 \) and \( J = \text{diag}(0.0081, 0.0081, 0.0142) \text{kg m}^2/\text{rad}^2 \). The unstructured forces and moments, \( \Delta_2 \) and \( \Delta_3 \), are modelled by sinusoidal signals with different phases for each channel and amplitudes equal to 0.5 and 0.1 respectively. The \( \delta_{\text{max}} \) is selected to be 1 which is equal to distance of 1m from the object.

In the simulation, the robot is assumed in hover condition and its initial position is at \( (2, -0.5, -5) \) with respect the to inertial frame and desired image features are obtained at \( (0, 0, -8) \). The observed target is assumed to move on a flat ground which follows a circle path with radius equal to 1m. The control gains, to satisfy conditions in (20) are set as \( k_1 = 1.3, k_2 = 2.8, k_3 = 14, k_4 = 20 \). Variables \( \epsilon_1(t), \epsilon_2(t) \) are selected to be constant and equal to 0.1. The yaw rotation is controlled to have zero velocity by the method presented in [11].

Simulation results are illustrated in Fig. 1. As expected, results show satisfactory tracking of the object and error signals and also the system states are bounded in the tracking mission and converge to a small compact set.

VI. CONCLUSION

In this paper, a robust image based visual servo controller has been developed for translational motions of a quadrotor helicopter flying on a moving target. A nonlinear controller based on backstepping approach is designed for full dynamics of the system. The controller is robust with respect to uncertainties in dynamics of the image features and also dynamics of the robot. Simulation results show satisfactory performance of the presented visual servo approach in both image space and Cartesian space of the robot when tracking a moving target and states of the system are uniformly ultimately bounded and converge to a small compact set.

Our future work is to design an observer based vision based controller to compensate lack of accuracy in linear velocity information of the robot. The future work is also devoted to verify the effectiveness of the proposed control algorithms by experiments on a real quadrotor.

REFERENCES