Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

Quadrotor Control Using Dual Camera Visual Feedback

Erdinç Altuğ, James P. Ostrowski, Camillo J. Taylor
GRASP Lab. University of Pennsylvania, Philadelphia, PA 19104, USA
E-mail: {erdinc, jpo, cjtaylor}@grasp.cis.upenn.edu

Abstract— **In this paper, a vision-based stabilization and output tracking control method for a four-rotor helicopter has been proposed. A novel 2 camera method has been described for estimating the full 6 DOF pose of the helicopter. This two camera system is consisting of a pan-tilt ground camera and an onboard camera. The pose estimation algorithm is compared in simulation to other methods (such as four point method, and a stereo method) and is shown to be less sensitive to feature detection errors on the image plane. The proposed pose estimation algorithm and nonlinear control techniques have been implemented on a remote controlled quadrotor helicopter.**

I. INTRODUCTION

The purpose of this study is to explore control methodologies and pose estimation algorithms that will make an unmanned aerial vehicle (UAV) autonomous. An autonomous UAV will be suitable for applications like search and rescue, surveillance and remote inspection. Rotary wing aerial vehicles have distinct advantages over conventional fixed wing aircrafts on surveillance and inspection tasks, since they can take-off/land in limited spaces and easily hover above the target. A ***quadrotor*** is a four rotor helicopter. Recent work in quadrotor design and control includes quadrotor [1], X4-Flyer [2], mesicopter [3] and hoverbot [4]. Also, related models for controlling the VTOL aircraft are studied by Hauser et al [5].

Quadrotor is an under-actuated, dynamic vehicle with four input forces and six output coordinates. Unlike regular helicopters that have variable pitch angle rotors, a quadrotor helicopter has four fixed pitch angle rotors. Advantages of using a multi-rotor helicopter are the increased payload capacity and high maneuverability. Disadvantages are the increased helicopter weight and increased energy consumption due to the extra motors. The basic motions of a quadrotor are generated by varying the rotor speeds of all four rotors, thereby changing the lift forces. The helicopter tilts towards the direction of slow spinning rotor, which enables acceleration along that direction. Therefore control of the tilt angles and the motion of the helicopter are closely related and estimation of orientation (roll and pitch) is critical. Spinning directions of the rotors are set to balance the moments, therefore eliminating the need for a tail rotor. This is also used to produce the desired yaw motions. A good controller should properly arrange the rotor speeds so that only the desired states change.

In order to create an autonomous UAV, precise knowledge of the helicopter position and orientation is needed. This info can be used to stabilize, hover the helicopter or for tracking an object. The pose estimation of a 3D robot has also been studied by [9], [10], [11], [12]. But in these papers, a single onboard camera has been used and the estimates were obtained by combining image data with readings from the inertial navigation systems, GPS or gyros. Our primary goal is to investigate the possibility of a purely vision-based controller on the quadrotor. Limited payload capacity does not permit the use of heavy navigation systems or GPS. Moreover the GPS does not work at indoor environments. One can still setup an indoor GPS system or use small navigation systems but, cost limits the use of these systems. This study utilizes a two camera system for pose estimation. Unlike previous work that either utilizes monocular views or stereo pairs, our two cameras are set to see each other. A ground camera that has pan-tilt capability, and an onboard camera are used to get accurate pose information. The proposed pose estimation algorithm is compared in simulation with other methods like a four-point algorithm [13], a state estimation algorithm [14] and a direct method that uses the area estimations of the blobs. A backstepping like controller [7], [8] shown in [1] has been implemented and shown effective in simulations of the dynamical quadrotor model. The proposed pose estimation algorithm and the control techniques have been implemented on a remote controlled, battery powered model helicopter.

II. HELICOPTER MODEL

The quadrotor helicopter model is shown in Figure 1. A body-fixed frame (**B**) is assumed to be at the center of gravity of the quadrotor, where the z-axis is pointing upwards. This body axis is related to the inertial frame (**O**) by a position vector p=(x,y,z) $\in$ **O** and a rotation matrix $R : O \rightarrow B$, where $R \in \mathbf{SO}(3)$. A ZYX Euler angle representation has been chosen to represent the rotations. It is composed of 3 Euler angles, (ϕ, θ, ψ), representing yaw, roll (rotation around y-axis) and pitch (rotation around x-axis), respectively.

A spinning rotor produces moment as well as thrust. Let F_i be the thrust and M_i be the moment generated by rotor i, that is spinning with rotational speed of w_i.

Let $V_b \in \mathbf{B}$ be the linear velocity in body-fixed frame and $w_b \in \mathbf{B}$ the angular velocity. Therefore the velocities will be

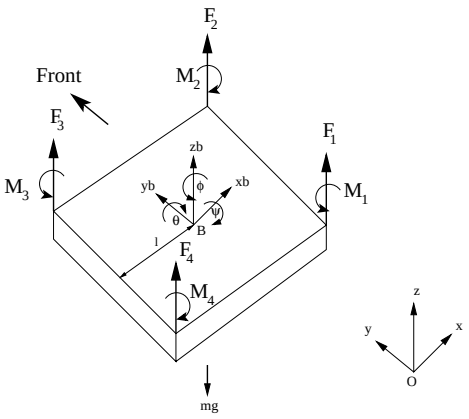

Fig. 1. 3D Quadrotor Model.

$$V_b = R^T \dot{p} \tag{1}$$

$$skew(w_b) = R^T \dot{R} \tag{2}$$

where $skew(w) \in \mathbf{so}(3)$ is the skew symmetric matrix of w. To represent the dynamics of the quadrotor, we can write the Newton-Euler equations as follows

$$m\dot{V}_b = F_{ext} - w_b \times mV_b \tag{3}$$

$$I_b \dot{w}_b = M_{ext} - w_b \times I_b w_b. \tag{4}$$

F_{ext} and M_{ext} are the external forces and moments on the body-fixed frame. I_b is the inertia matrix, and m is the mass of the helicopter.

Drag on a moving object [17] is given by $Drag = \frac{1}{2} C_d \rho v^2 A$, in which ρ is the density of air, A is the frontal area, C_d is the drag coefficient, and V is the velocity. Assuming constant ρ, the constants at the above equation can be combined to form C, which simplifies drag to $Drag = Cv^2$.

The force generated by a rotor [17] which is spinning with rotational velocity of w is given by $F = bL = \frac{\rho}{4} w^2 R^3 abc(\theta_t - \phi_t)$, where b is the number of blades on a rotor, θ_t is the pitch at the blade tip, ϕ_t is the inflow angle at the tip. By combining the constant terms as constant variable D, this equation simplifies to $F_i = D{w_i}^2$.

Therefore F_{ext} and M_{ext} will be

$$F_{ext} = -C_x \dot{x}^2 \hat{i} - C_y \dot{y}^2 \hat{j} + (T - C_z \dot{z}^2)\hat{k} - R \cdot mg\hat{k} \tag{5}$$

$$M_{ext} = M_x \hat{i} + M_y \hat{j} + M_z \hat{k} \tag{6}$$

where C_x, C_y, C_z are the drag coefficients along x, y and z axes, respectively. T is the total thrust and M_x, M_y, M_z are the moments generated by the rotors. The relation of thrust and moments to the rotational velocities of rotors is given as follows

$$\begin{pmatrix} T \\ M_x \\ M_y \\ M_z \end{pmatrix} = \begin{pmatrix} D & D & D & D \\ -Dl & Dl & Dl & -Dl \\ -Dl & -Dl & Dl & Dl \\ CD & -CD & CD & -CD \end{pmatrix} \begin{pmatrix} {w_1}^2 \\ {w_2}^2 \\ {w_3}^2 \\ {w_4}^2 \end{pmatrix}. \tag{7}$$

The above matrix $M \in \mathbb{R}^{4\times4}$ is full rank for $l, C, D \neq 0$. The rotational velocity of rotor i (w_i), can be related to the torque of motor i (τ_i) as

$$\tau_i = I_r \dot{w}_i + K{w_i}^2 \tag{8}$$

where I_r is the rotational inertia of rotor i, K is the reactive torque due to the drag terms.

Motor torques τ_i should be selected to produce the desired rotor velocities (w_i) in Equation 8, which will change the external forces and moments in Equations 5 and 6. This will produce the desired body velocities and accelerations in Equations 3 and 4.

III. CONTROL OF A QUADROTOR HELICOPTER

A controller should pick suitable rotor speeds w_i for the desired body accelerations. Let's define the control inputs to be

$$\begin{aligned} u_1 &= (F_1 + F_2 + F_3 + F_4) \\ u_2 &= l(-F_1 + F_2 + F_3 - F_4) \\ u_3 &= l(-F_1 - F_2 + F_3 + F_4) \\ u_4 &= C(F_1 - F_2 + F_3 - F_4). \end{aligned} \tag{9}$$

C is the force-to-moment scaling factor. The u_1 represents a total thrust on the body in the z-axis, u_2 and u_3 are the pitch and roll inputs and u_4 is a yawing moment. Backstepping controllers [7] are useful when some states are controlled through other states. Since motions along the x and y axes are related to tilt angles θ and ψ respectively, backstepping controllers given in [1] can be used to control tilt angles enabling the precise control of the x and y motions (inputs u_2 and u_3). The altitude and the yaw, can be controlled by PD controllers

$$\begin{aligned} u_1 &= \frac{g + K_{p1}(z_d - z) + K_{d1}(\dot{z_d} - \dot{z})}{\cos\theta \cos\psi} \\ u_4 &= K_{p2}(\phi_d - \phi) + K_{d2}(\dot{\phi}_d - \dot{\phi}). \end{aligned} \tag{10}$$

The proposed model and the controllers were tested in simulations. In Figure 2, the quadrotor moves from (30,40,150) to the origin with an initial yaw of 20 degrees and zero tilt angles. Note the tilt-up motions of the quadrotor towards the end of the simulation, performed to slow down and reach the origin with zero velocity. Figure 3 shows the motion of the quadrotor during this simulation.

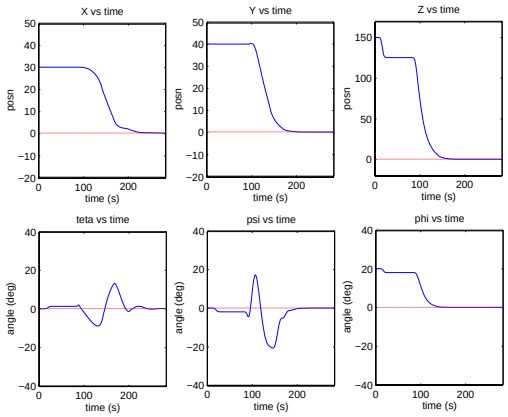

Fig. 2. Quadrotor simulation results.

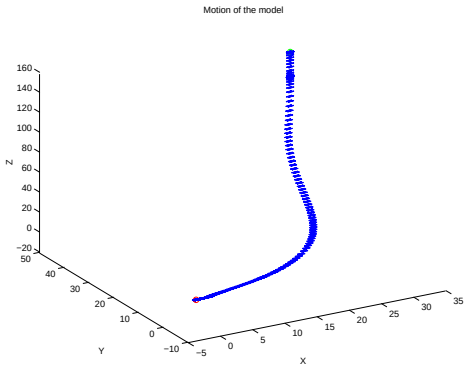

Fig. 3. The path of the quadrotor.

IV. POSE ESTIMATION

For autonomous helicopters, estimation of motion (relative position, orientation and velocities) is important for surveillance and remote inspection tasks or taking off / landing from a site. For our project, the goal is to obtain the pose from vision rather than complex and heavy navigation systems or GPS. For this purpose, a pair of color cameras track multi-color blobs located under the helicopter and on the ground. A blob tracking algorithm is used to determine the positions and areas of the blobs on the image planes. Therefore the purpose of the pose estimation algorithm is to obtain (*x*,*y*,*z*) positions, tilt angles (θ, ψ), the yaw angle (ϕ) and the velocities of the helicopter, in real-time relative to the ground camera frame. These can be represented as the Rotation Matrix, $R \in \mathbf{SO}(3)$, defining the body-fixed frame of the helicopter with respect to the ground camera frame, where $R^T R = I$, $det(R) = 1$, and the relative position vector $p \in \mathbf{R}^3$, which is the position of the helicopter with respect to the ground camera.

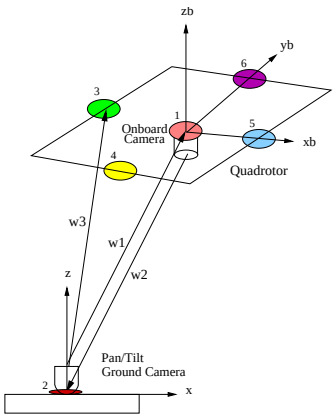

Fig. 4. Two camera pose estimation method.

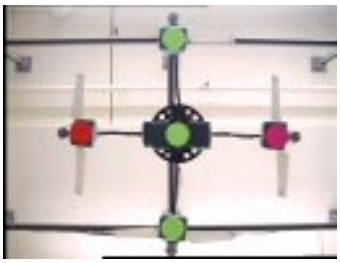

Fig. 5. Quadrotor tracking with a camera.

A. Two Camera Pose Estimation Method

The two camera pose estimation method uses a pan-tilt ground camera and an onboard camera. Previous work on vision-based pose estimation utilizes monocular views or stereo pairs. Our two camera pose estimation method involves the use of two cameras that are set to see each other as shown in Figure 4. This method is especially useful for autonomous taking off or landing. Colored blobs of 2.5 cm radius are attached to the bottom of the quadrotor and to the ground camera as shown in Figure 5. Tracking two blobs on the quadrotor image plane and one blob on the ground image frame is found to be enough for accurate pose estimation. To minimize the error as much as possible, five blobs are placed on the quadrotor and a single blob is located on the ground camera. The blob tracking algorithm tracks the blobs and returns image values (u_i, v_i) for $i = 1 \cdots 6$.

The cameras have matrices of intrinsic parameters, A_1 and A_2. The unit vector $w_i \in \mathbf{R}^3$ from each camera to the blobs can be found as

$$\begin{aligned} w_i &= inv(A_1).[u_i \quad v_i \quad 1]', \quad w_i = w_i/norm(w_i) \\ &\quad for \quad i = 1,3,4,5,6 \\ w_2 &= inv(A_2).[u_2 \quad v_2 \quad 1]', \quad w_2 = w_2/norm(w_2) \end{aligned} \tag{11}$$

Let L_a be the vector pointing from blob-1 to blob-3 in Figure 4. Vectors w_1 and w_3 are related by

$$\lambda_3 w_3 = \lambda_1 w_1 + RL_a \tag{12}$$

where λ_1 and λ_3 are unknown scalars. Taking the cross product with w_3 gives

$$\lambda_1 (w_3 \times w_1) = RL_a \times w_3. \tag{13}$$

This can be rewritten as

$$(w_3 \times w_1) \times (RL_a \times w_3) = 0. \tag{14}$$

Let the rotation matrix R be composed of two rotations: the rotation of θ degrees around the vector formed by the cross product of w_1 and w_2 and the rotation of α degrees around w_1. In other words:

$$R = Rot(w_1 \times w_2, \theta) \cdot Rot(w_1, \alpha) \tag{15}$$

where $Rot(a,b)$ means the rotation of b degrees around the unit vector a. The value of θ can be found from the dot product of vectors w_1 and w_2.

$$\theta = acos(w_1 \cdot w_2) \tag{16}$$

The only unknown is the angle α. Let M be a matrix described as:

$$M = (w_3 \times w_1) \times (w_3 \times (R(w_1 \times w_2, \theta))). \tag{17}$$

Using Rodrigues' formula [15], Equation 14 can be simplified to

$$M \cdot L_a + \sin\alpha \cdot M\widehat{w_1} \cdot L_a + (1 - \cos\alpha) \cdot M \cdot (\widehat{w_1})^2 \cdot L_a = 0. \tag{18}$$

This is a set of three equations in the form of $A\cos\alpha + B\sin\alpha = C$, which can be solved by

$$\alpha = \arcsin \frac{B \cdot C \pm \sqrt{(B^2 \cdot C^2 - (A^2 + B^2) \cdot (C^2 - A^2))}}{A^2 + B^2} \tag{19}$$

One problem here is that $\alpha \in [\pi/2, -\pi/2]$, because of the arcsin function. Therefore one must check the unit vector formed by two blobs to find the heading, and pick the correct α value.

Thus, the estimated rotation matrix will be $R = Rot(w_1 \times w_2, \theta) \cdot Rot(w_1, \alpha)$. Euler angles ($\phi$, θ, ψ) defining the orientation of the quadrotor can be obtained from rotation matrix, R.

In order to find the relative position of the helicopter with respect to the inertial frame located at the ground camera frame, we need to find scalars λ_i, for $i = 1 \cdots 6$. λ_1 can be found using Equation 12. The other λ_i values $(\lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6)$ can be found from

$$\lambda_i w_i = \lambda_1 w_1 + RL_i. \tag{20}$$

L_i is the position vector of i^{th} blob in body-fixed frame. To reduce the errors, λ_i values are normalized using the blob separation, L.

The center of the quadrotor will be

$$\begin{aligned} X &= (\lambda_3 w_3(1) + \lambda_4 w_4(1) + \lambda_5 w_5(1) + \lambda_6 w_6(1))/4 \\ Y &= (\lambda_3 w_3(2) + \lambda_4 w_4(2) + \lambda_5 w_5(2) + \lambda_6 w_6(2))/4 \\ Z &= (\lambda_3 w_3(3) + \lambda_4 w_4(3) + \lambda_5 w_5(3) + \lambda_6 w_6(3))/4. \end{aligned} \tag{21}$$

B. Comparing the Pose Estimation Methods

The proposed two camera pose estimation method is compared to other methods using a Matlab simulation. Other methods used were a four-point algorithm [13], a state estimation algorithm [14], a direct method that uses the area estimations of the blobs, and a stereo pose estimation method that uses two ground cameras that are separated by a distance d.

The errors are calculated using angular and positional distances, given as

$$\begin{aligned} e_{ang} &= \| \, log(R^{-1} \cdot R^{est}) \, \| \\ e_{pos} &= \| \, p - p^{est} \, \| \, . \end{aligned} \tag{22}$$

R^{est} and p^{est} are the estimated rotational matrix and the position vector. Angular error is the amount of rotation about a unit vector that transfers R to R^{est}.

Figures 6, 7 and 8 show the motion of the quadrotor and the pose estimation errors. Quadrotor moves from the point (22, 22, 104) to (60, 60, 180) cm., while (θ, ψ, ϕ) changes from (0.7, 0.9, 2) to (14, 18, 40) degrees. A random error up to 5 pixels were added on image values. The blob areas were also added a random error of magnitude ± 2. The comparison of the pose estimation methods and the average angular and positional errors are given on Table 1.

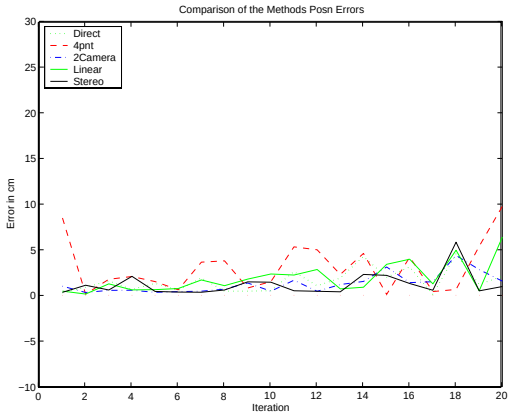

Fig. 6. Comparison of the Position Estimation Errors.

It can be seen from the plots and Table 1 that, the estimation of orientation is more sensitive to errors on the image plane than the position estimation. The poor orientation estimate of the direct method is because of the blob areas which are subject to the random noise. For the stereo method, the value of the baseline is important for

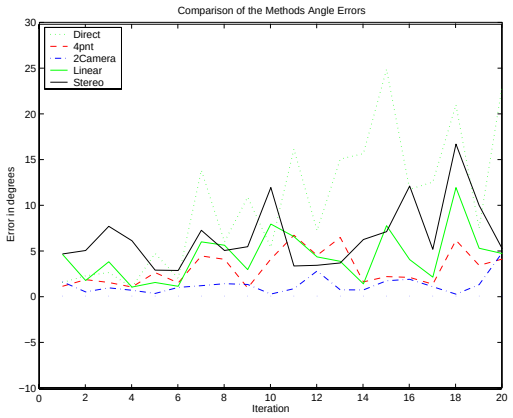

Fig. 7. Comparison of the Orientation Estimation Errors.

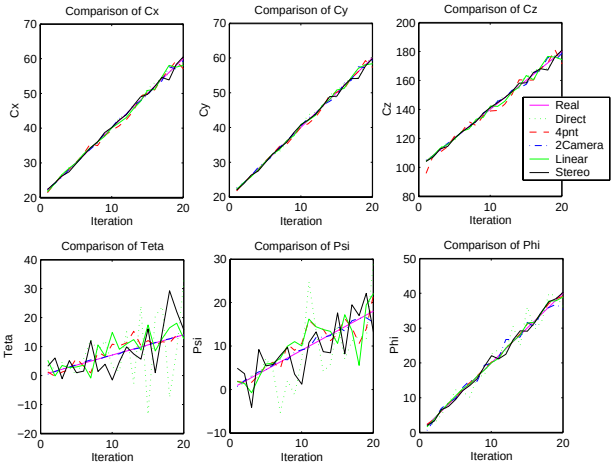

Fig. 8. Estimated helicopter positions and orientation angles.

pose estimation. Figure 9 shows the effects of the baseline distance change on the estimation. As the baseline distance approaches 3 times the distance between blobs ($6L$), the stereo method appears to be giving good estimates. The need for a large baseline for stereo pairs is the drawback of the stereo method. The average error of the two camera method is 1.22 degrees for angular estimation and 1.26 cm. for positional estimation, so we can conclude that the two camera method is more effective comparing to the other methods. The two camera method has other advantages, such as its ability to perform even when some blobs are lost. But for other methods the loss of a single blob will result bad pose estimates.

V. EXPERIMENTS

The proposed controllers and the pose estimation algorithms have been implemented on a remote-controlled battery-powered helicopter shown in Figure 10a. It is a commercially available model helicopter called HMX-4. It is about 0.7 kg, 76 cm long between rotor tips and has

TABLE I

COMPARISON OF THE POSE ESTIMATION METHODS.

Method	Angular E. (deg.)	Positional E. (cm.)
Direct M.	10.2166	1.5575
4 Pnt. M.	3.0429	3.0807
2 Camera M.	1.2232	1.2668
Linear M.	4.3700	1.8731
Stereo M.	6.5467	1.1681

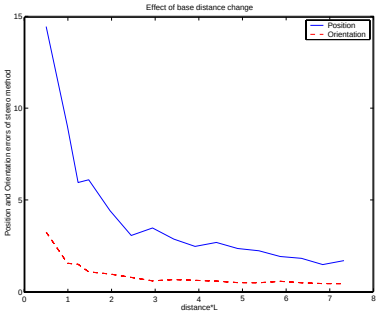

Fig. 9. Effects of baseline distance change on pose estimation at the Stereo Method.

about 3 minutes flight time. This helicopter has three gyros on board to stabilize itself. An experimental setup shown in Figure 10b was prepared to prevent the helicopter from moving too much on the x-y plane, while enabling it to turn and ascend/descend freely. Vision based stabilization experiments were performed using the two camera pose estimation method. In these experiments two separate computers were used. Each camera was connected to separate computers which were responsible for performing blob tracking. PC-1 did the computation for the onboard camera and sent the information to PC-2 via the network. PC-2 was responsible for the ground camera and for the calculation of the control signals. These signals were then sent to the helicopter with a remote control device that uses the parallel port.

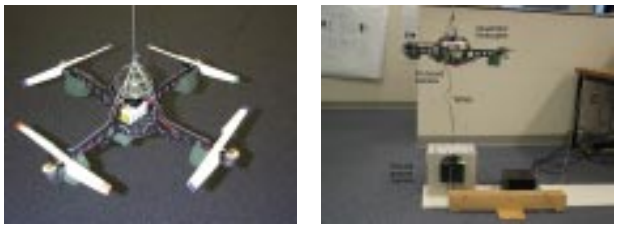

Fig. 10. a) Quadrotor Helicopter, b) Experimental Setup.

Controllers described in [1] were implemented for the experiment. Figure 11 shows the results of this experiment using the two camera pose estimation method, where height, x, y and yaw angle are being controlled. The mean and standard deviation are found to be 129 cm and 13.4 cm

for z, 5.86 degrees and 17.2 degrees for ϕ respectively. The results from the plots show that the proposed controllers do an acceptable job despite the pose estimation errors.

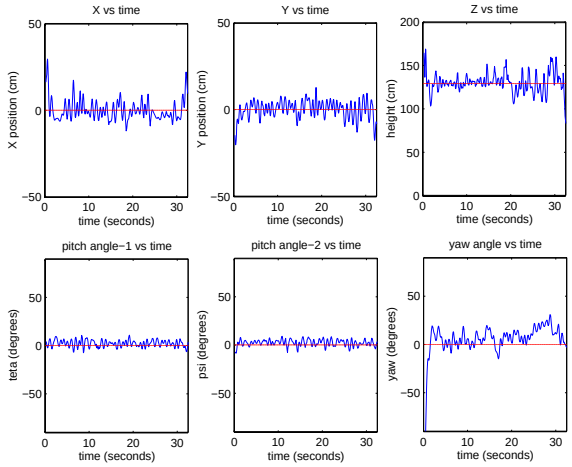

Fig. 11. The results of the height x, y and yaw control experiment with two camera pose estimation method.

VI. CONCLUSIONS AND FUTURE WORK

We have presented a novel two camera method for pose estimation. The method has been compared to other pose estimation algorithms and shown to be more effective especially when there are errors on the image plane. Backstepping controllers have been used to stabilize and perform output tracking control. Simulations performed on Matlab Simulink show the ability of the controller to perform output tracking control even when there are errors on state estimates. The proposed controllers and the pose estimation method have been implemented on a remote-control, battery-powered model helicopter. Initial experiments on a tethered system showed that the vision-based control is effective in controlling the helicopter. Our future work will include placing the ground camera on top of a mobile robot and enabling take-off/landing from a mobile robot. Such functionalities will be useful for inspection, chase and other ground-air cooperation tasks.

VII. ACKNOWLEDGMENTS

We gratefully acknowledge support from the Republic of Turkey Ministry of National Education, NSF and DARPA.

VIII. REFERENCES

[1] E. Altuğ, J. P. Ostrowski, R. Mahony, *Control of a Quadrotor Helicopter using Visual Feedback*, Proceedings of the IEEE International Conference on Robotics and Automation, Washington, D.C., May 2002, pp. 72-77.

[2] T. Hamel, R. Mahony, A. Chriette, *Visual Servo trajectory tracking for a four rotor VTOL aerial vehicle*, Proceedings of the 2002 IEEE International Conference on Robotics and Automation, Washington, D.C., May 2002, pp. 2781-2786.

[3] I. Kroo, F. Printz, *Mesicopter Project*, Stanford University, http://aero.stanford.edu/mesicopter.

[4] J. Borenstein, *Hoverbot Project*, Univ. of Michigan, www-personal.engin.umich.edu/∼johannb/hoverbot.ht

[5] J. Hauser, S. Sastry, G. Meyer, *Nonlinear Control Design for Slightly non-minimum Phase Systems: Application to V/STOL Aircraft*, Automatica, vol 28, No:4, 1992, pp. 665-679.

[6] P. Martin, S. Devasia, B. Paden . *A different look at output tracking: Control of a VTOL aircraft*, Automatica No:1, 32, 1996, pp. 101-107.

[7] S. Sastry, *Nonlinear Systems; Analysis, Stability and Control*, Springer-Verlag, 1999.

[8] T. Hamel, R. Mahony, *Visual servoing of a class of under-actuated dynamic rigid-body systems*, Proceedings of the 39th IEEE Conference on Decision and Control, 2000.

[9] O. Amidi, *An Autonomous Vision Guided Helicopter*, Ph.D. Thesis, Carnegie Mellon University, August 1996.

[10] H. Zhang, *Motion Control for Dynamic mobile robots*, Ph.D. Thesis, University of Pennsylvania, June 2000.

[11] H. Shim, *Hierarchical Flight Control System Synthesis for Rotorcraft-based Unmanned Aerial Vehicles*, Ph.D. Thesis, University of California, Berkeley, Fall 2000.

[12] Y. Ma, J. Kosecka, S. Sastry, *Optimal Motion from image Sequences: A Riemannian viewpoint*, Research Report, University of California Berkeley, UCB/ER/M98/37.

[13] A. Ansar, D. Rodrigues, J. Desai, K. Daniilidis, V. Kumar, M. Campos, *Visual and Haptic Collaborative Tele-presence*, Computer and Graphics, Special Issue on Mixed Realities Beyond Convention, Oct 2001.

[14] C. S. Sharp, O. Shakernia, S. S. Sastry, *A Vision System for Landing an Unmanned Aerial Vehicle*, IEEE Conference on Robotics and Automation, 2001.

[15] R. Murray, Z. Li, S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.

[16] E. Trucco, A. Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice-Hall, 1998, pp. 322-325.

[17] R. Prouty, *Helicopter Performance, Stability, and Control*, Krieger Publishing Company, June 1995.