SIMULATION OF TRACKING A MOVING TARGET WITH A CAMERA ON A PAN/TILT HEAD

Daniel Sotingco
May 12, 2011

INTRODUCTION

This project simulates the tracking of a moving target with a camera on a pan/tilt head. The camera is located at the origin of the system, with a target positioned somewhere around it in three-dimensional space. The goal of the controller is to keep the camera pointed at the target, such that it always appears at the center of the image taken by the camera. A set-point controller, adaptive controller for trajectory tracking, and set-point controller with vision were implemented in simulation for the system.

EQUATIONS OF MOTION

A diagram of the system is shown below in Figure 1. The angles q_1 and q_2 refer to the pan and tilt angles of the camera. The coordinates of the target, shown as a red ball below, are specified in cylindrical coordinates, with ρ being the Euclidean distance from the vertical *z*-axis to the target, ψ being the angle from the reference direction in the horizontal *xy* plane to the ball (essentially, it is the "pan angle" of the target), and *z* being the distance from the *xy* plane to the target.

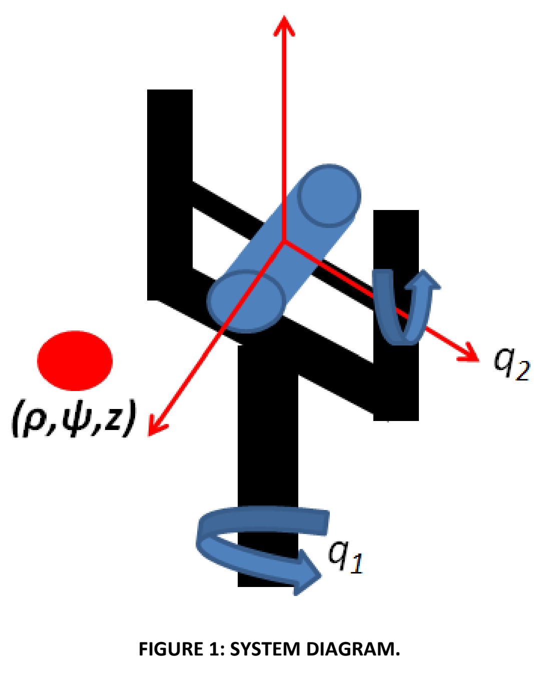

FIGURE 1: SYSTEM DIAGRAM.

The governing equations of the pan/tilt system without vision are obtained from (1) and are based on the dynamics of the Fast-Eye Gimbal (FEG). These governing equations are shown below.

$$H\underline{\ddot{q}} + C\underline{\dot{q}} + D\underline{\dot{q}} + G = \underline{\tau}$$

$$H = \begin{bmatrix} 2J_{12}\cos q_2 \sin q_2 + J_{11} + I_{22+}(m_2 d^2 + J_{22} - J_{11})\cos^2 q_2 & h_1 \\ h_1 & m_2 d^2 + J_{33} \end{bmatrix}$$

$$h_1 = J_{13}\sin q_2 + J_{23}\cos q_2$$

$$C = \begin{bmatrix} -c_1\dot{q}_2 & -c_1\dot{q}_1 + (J_{13}\cos q_2 - J_{23}\sin q_2)\dot{q}_2 \\ c_1\dot{q}_1 & 0 \end{bmatrix}$$

$$c_1 = (1 - 2\cos^2 \mathrm{q}_2)J_{12} + (m_2 d^2 + J_{22} - J_{11})\cos q_2 \sin q_2$$

$$G = \begin{bmatrix} 0 \\ m_2 g d \cos q_2 \end{bmatrix}$$

Figure 2 shows a plot of the pan and tilt angles over time to verify that the equations make sense, and indeed, after some initial oscillations, the pan angle settles at zero, while the tilt angle settles at an angle of *–π/2* (essentially pointing downwards due to the absence of any motor torque).

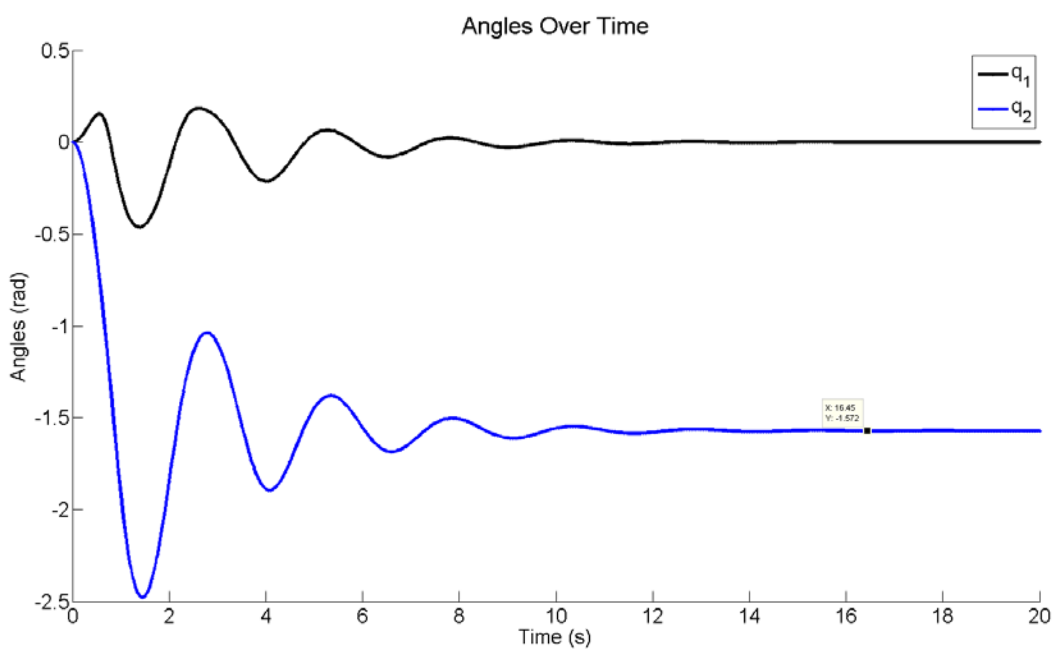

FIGURE 2: PAN AND TILT ANGLES FOR UNCONTROLLED SYSTEM.

Shown in Figure 3 are the geometries used to determine the transformations from the camera angles and target coordinates to the position of the target in image space.

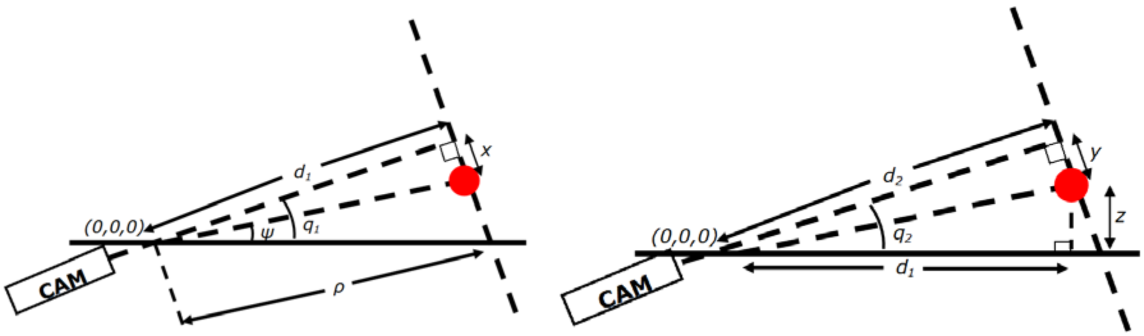

FIGURE 3: PANNING GEOMETRY (LEFT) AND TILTING GEOMETRY (RIGHT).

The equations below provide the location of the target in the camera's image (in pixels), given the camera's focal length f, scaling constant β, the camera angles q_1 and q_2, and the target location (ρ,ψ,z). The lengths $d_1(t)$ and $d_2(t)$ are intermediate variables: d_1 is the distance from the origin to a line passing through the ball perpendicular to the camera axis, and d_2 is the perpendicular distance from the origin to the plane of the ball in 3D space.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} \left(\dfrac{f\beta}{d_1 - f}\right) d_1 \tan(q_1 - \varphi) \\ \left(\dfrac{f\beta}{d_2 - f}\right) d_2 \tan\big(q_2 - \operatorname{atan}(z/\rho)\big) \end{bmatrix}$$

$$d_1 = \rho\cos(q_1 - \varphi)$$

$$d_2 = \sqrt{{d_1}^2 + z^2}\cos\big(q_2 - \operatorname{atan}(z/\rho)\big)$$

The equations above can be differentiated with respect to time to obtain velocities in the image plane.

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} \dfrac{f\beta}{d_1 - f}\left(\dfrac{-f\beta}{(d_1 - f)^2}\right)\{\dot{d}_1\tan(q_1 - \varphi) + d_1(\dot{q}_1 - \dot{\varphi})\sec^2(q_1 - \varphi)\} \\ \dfrac{f\beta}{d_2 - f}\left(\dfrac{-f\beta}{(d_2 - f)^2}\right)\left\{\dot{d}_2\tan\big(q_2 - \operatorname{atan}(z/\rho)\big) + d_2\left(\dot{q}_2 - \dfrac{\rho\dot{z} - z\dot{\rho}}{\rho^2 + z^2}\right)\sec^2\big(q_2 - \operatorname{atan}(z/\rho)\big)\right\} \end{bmatrix}$$

Attempts were made to express the above equation as:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = J_I J \underline{\dot{q}}$$

Trigonometric identities and Taylor series approximations were used to break up the equation; however, the Jacobian matrices were not successfully extracted, and so control was done without knowledge of the structure of these Jacobian matrices.

PD Set-Point Control

PD control to keep the camera pointed at a desired set of angles $[q_{1d}\ q_{2d}]^T$ can be accomplished with the following control law:

$$\tau = -K_P\underline{\tilde{q}} - K_D\underline{\dot{q}}$$

This has the effect of attaching a virtual spring to the desired angles and virtual dampers in the joints. However, because there is no integral term, there will be some steady-state error in q_2, which is affected by gravity. This is shown in Figure 4.

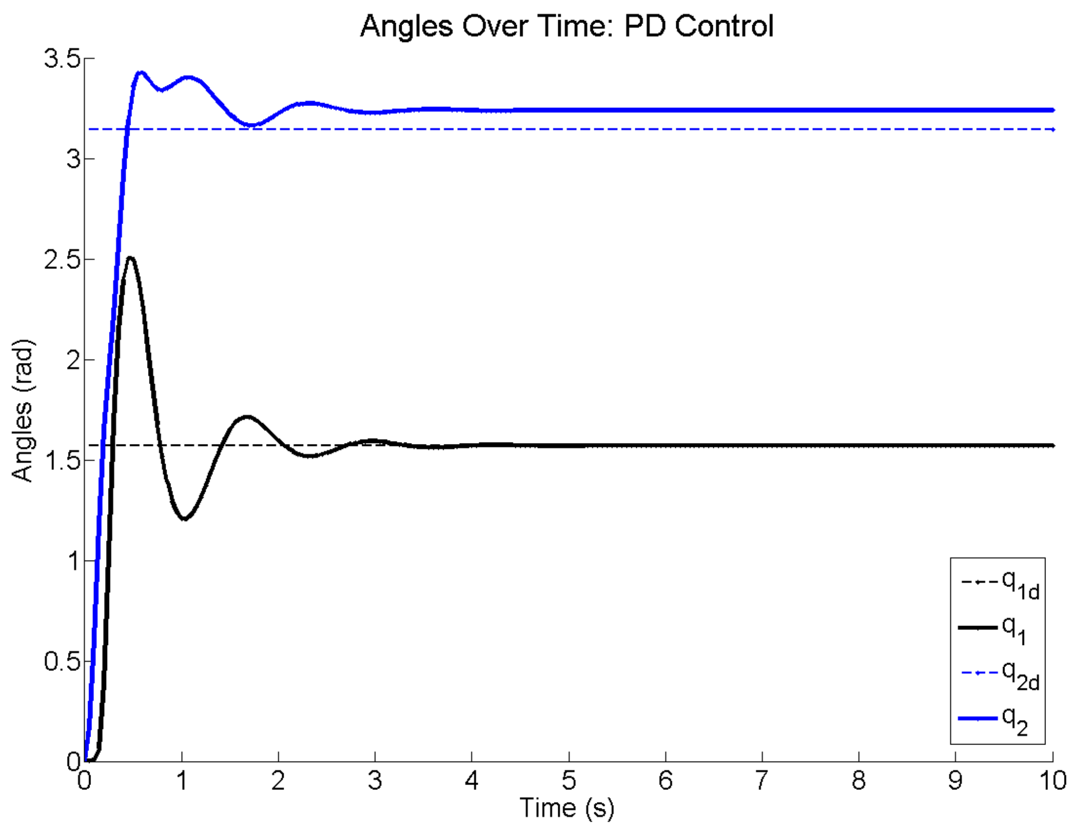

FIGURE 4: SET-POINT CONTROL WITH PD.

TARGET TRACKING WITH VISION AND PID

Here, the vision system is incorporated into the controller, with the goal of keeping the target at the center of the image plane. Ideally, the control law would look like:

$$\underline{\tau} = -J^T J_i^{\ T} K_P \underline{\tilde{x}_i} - K_D\underline{\dot{q}} + \underline{g}(\underline{q})$$

However, in this simulation, it is assumed that the controller only knows $[q_1\ q_2]^T$ and $[x_i\ y_i]^T$. Therefore, a modified controller is used in the simulation:

$$\underline{\tau} = -K_P\underline{\tilde{x}_i} - K_D\underline{\dot{q}} - K_i \int \underline{\tilde{x}_i}\, dt$$

As the control law shows, PID is used to deal with the steady-state error that would have been induced by gravity.

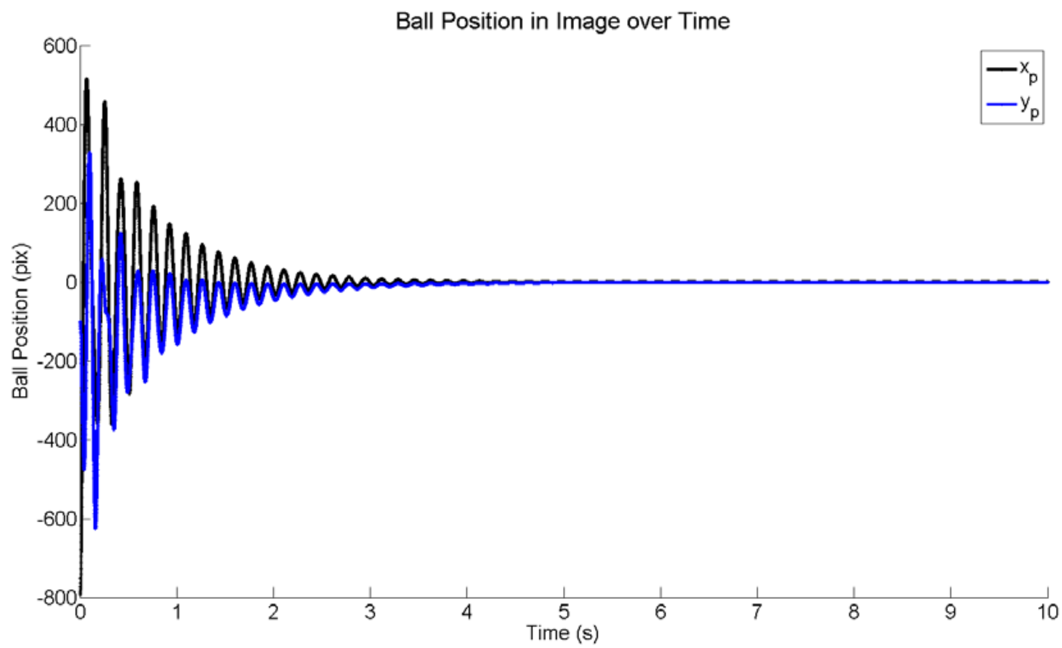

FIGURE 5: PID WITH VISION.

ADAPTIVE TRAJECTORY FOLLOWING

For fast trajectory following over time, a more sophisticated control scheme is needed. Here, adaptive control is used. The control law and adaption law are given below.

$$\underline{\tau} = Y\underline{\hat{a}} - K_D\underline{s}$$

$$\underline{\dot{\hat{a}}} = -PY^T\underline{s}$$

$$\underline{a} = [J_{12} \quad J_{11} + J_{22} \quad m_2 d^2 + J_{22} - J_{11} \quad J_{13} \quad J_{23} \quad m_2 d^2 + J_{33} \quad m_2 g d \quad D_{11} \quad D_{22}]^T$$

Simulated results of the pan/tilt head following sinusoidal trajectories are shown below in Figure 6. As shown, the adaptive control allows for close trajectory following, despite uncertainty in many of the system parameters.

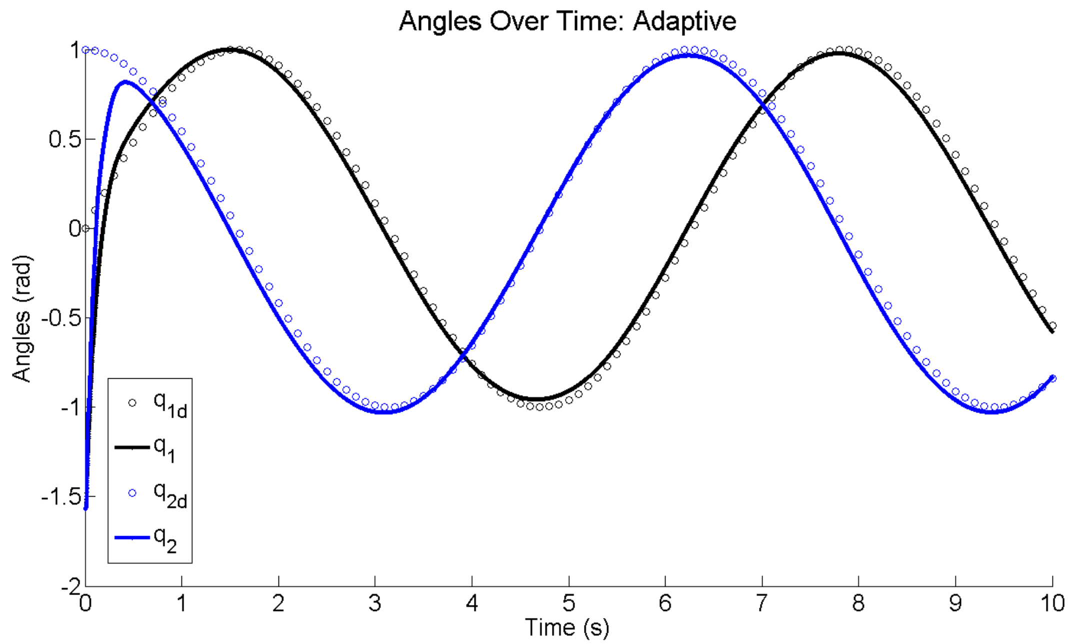

FIGURE 6: ADAPTIVE CONTROL OF PAN AND TILT ANGLES.

CONCLUSION

The equations of motion for a pan/tilt system with vision for target tracking were derived. Set-point control and adaptive control were then successfully implemented in simulation for control of the pan and tilt angles. PID was used in conjunction with vision to implement set-point control in the image space.

Because of the difficulty of expressing the equations of motion (with vision incorporated) using Jacobian matrices, adaptive control was not successfully implemented for the trajectory tracking with vision. An attempt was made at accomplishing this by using the identity matrix for *Y* (thereby lumping all terms together as unknown); however, this did not work, and so will not be described further. Implementing this successfully would be a future direction for this project.

BIBLIOGRAPHY

1. **Hong, Won.** *Adaptive Control of a Two Axis Camera Gimbal.* 1994.

2. *Adaptive Vision and Force Tracking Control for Robots with Constraint Uncertainty.* **Chien Chern Cheah, Saing Paul Hou, Yu Zhao, Jean-Jacques E. Slotine.** 3, s.l. : IEEE/ASME Transactions on Mechatronics, 2010, Vol. 15.

3. *Adaptive Vision based Tracking Control of Robots with Uncertainty in Depth Information.* **C.C. Cheah, C. Liu, J.J.E. Slotine.** Roma, Italy : IEEE International Conference on Robotics and Automation, 2007.