MAE 5810: Robot Perception
Mideterm Project Report

MODEL AND SIMULATION OF PINHOLE CAMERA

November 23, 2021

Haiyang Yu
Department of Electrical & Computer Engineering
College of Engineering
Cornell University
hy397@cornell.edu

1 Abstract

In this project, I derived the mathematical model of pan-tilt camera angular transformation and built its dynamic model with multiple constrains. Field of view is used to represent the camera model mathematical model. In addition, a simulation framework is build to simulate the controlling process of the camera. Finally, two target-tracking tasks are used as testing demos for this model.

2 Introduction

A pinhole camera is a simple camera with a small pinhole used to receive light and present the image on the photo. During the modeling process, the size of pinhole is sometimes ignored. In other words, the aperture is simplified to a point while considering optical problems. Pinhole camera is the simplest camera model. Taking advantage to its simplicity, Pinhole camera model is popularly used for robot perception research is regard as a reasonable description for camera-based systems. An example of Pinhole camera is shown in figure 1.

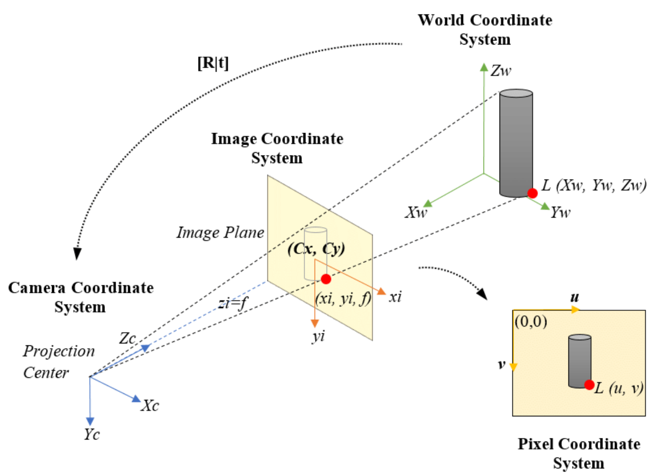


Figure 1: Pinhole camera model[1]

In this project, I will model a controllable pan-tilt (PT) camera based on a pinhole camera model. In this model, the camera is assumed to be fixed to the ceiling and its yaw (ψ) and roll (Φ) angles are controllable. The goal of the camera is tracking a moving target on the floor of the room. The following assumptions are made to simplify this project:

- The focal length λ and the size of virtual image plane are known.
- The camera lens is symmetric about an optical axis.

- The measuring process is noise-free.

3 Model Construction

3.1 Coordinate system conversion

As illustrated in Figure 2, the PT camera is fix on a point and can pan and tilt a certain angle. To construct a camera-based coordinate, the ground coordinate is first moved from coordinate origin to the camera position. After that, it rotates ψ and Φ along the z-axis and x-axis, respectively.

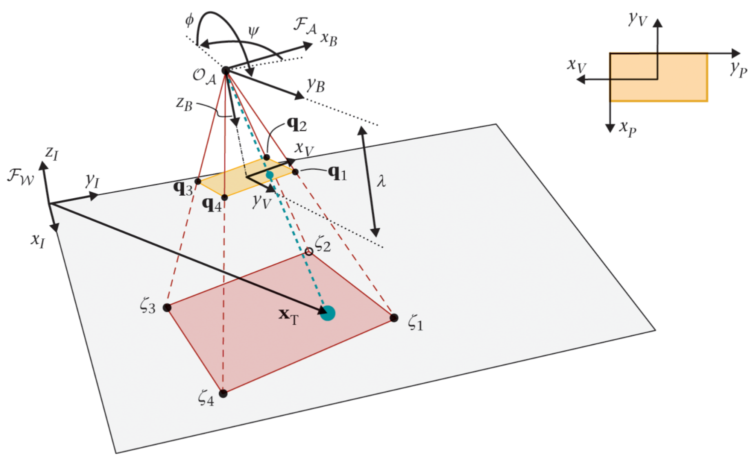


Figure 2: Pinhole camera model coordinate frames [2]

The moving coordinate conversion during the moving stage follows the function below:

$$\boldsymbol{x_c} = \boldsymbol{x_1} + \boldsymbol{x_b} \tag{1}$$

Where x_c is the internal position vector (in $\mathcal{F}_W$), x_1 is the position vector in the coordinate fixed at the camera location. $\boldsymbol{x_b}$ is the location of camera in the internal coordinate.

During the rotating process, the coordinate transition is based on Euler rotation matrices.

$$\boldsymbol{x_1} = R_\phi R_\psi \boldsymbol{x_2} \tag{2}$$

Where x_2 is the position corresponding to the camera coordinate $\mathcal{F}_{\mathcal{A}}$, R_ϕ and R_ψ are defined as Euler rotation matrices:

$$R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{bmatrix} \tag{3}$$

$$R_\psi = \begin{bmatrix} cos\psi & sin\psi & 0 \\ -sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

According to the functions above, we can compute the location of a target on the floor based on its position on the image frame. Assume the target projection is located at (x_a, y_a) on the image, since we already know the focal length of the camera λ. The location of projection point can be represented as $\boldsymbol{x_2} = [x_a, y_a, \lambda]^T$ in the $\mathcal{F_A}$. Its coordinate in the world coordinate system is computed by:

$$\boldsymbol{x_c} = R_\phi R_\psi \boldsymbol{x_2} + \boldsymbol{x_b} \tag{5}$$

Where $\boldsymbol{x_c} = [x_c, y_c, z_c]$ is coordinates in the world coordinate system, $\boldsymbol{x_b} = [x_b, y_b, z_b]$ is the position of the camera. Since we assume the target is moving on the ground, assume its real location is $\boldsymbol{x_r} = [x_r, y_r, 0]$, we can calculate x_r and y_r as below

$$\begin{aligned} x_r &= x_b + \frac{z_b}{z_b - z_c}(x_c - x_b) \\ y_r &= y_b + \frac{z_b}{z_b - z_c}(y_c - y_b) \end{aligned} \tag{6}$$

In conclusion, based on the deduction above, we can get the target position (x_r, y_r) if we have its projection location (x_a, y_a).

3.2 Camera Kinematic model

Assume the PT camera is driven by two motors which power the pan movement and tilt movement. The accelerations of pan and tilt angle are controllable. The camera kinematic equation can be expressed as state space function as below.

$$\boldsymbol{s}(k+1) = A\boldsymbol{s}(k) + B\boldsymbol{u}(k) \tag{7}$$

Where, $s = [\psi, \phi, \dot{\psi}, \dot{\phi}]^T$, $u = [\ddot{\psi}, \ddot{\phi}]$, b_1 and b_2 are motor parameters.

$$A = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ b_1 & 0 \\ 0 & b_2 \end{bmatrix} \tag{8}$$

Taking the motor constrains and physical bounds into account, the system should obey constrains as below.

$$\begin{cases} b_1 \leq s \leq b_2 \\ |u| \leq 1_2 \end{cases} \tag{9}$$

Where $b_1 = [0, \pi/2, -\dot{\psi}_{max}, -\dot{\phi}_{max}]^T$, $b_2 = [2\pi, \pi, \dot{\psi}_{max}, \dot{\phi}_{max}]^T$,$1_2 = [1, 1]^T$

4 Visualization and Simulation

4.1 Field of View visualization

In the visualization part, parameters are listed as follows:

Parameter	Description	Value(units)
x_b	X-cordinate of the camera	$1(m)$
y_b	Y-cordinate of the camera	$1(m)$
z_b	Z-cordinate of the camera	$6(m)$
λ	focal length	$50(mm)$
w	width of virtual image plane	$3.6(mm)$
h	height of virtual image plane	$2.7(mm)$

Table 1: Information of camera

Based on these camera parameters, a sample camera field of view(FOV) is illustrated in Figure 3 , where ψ and ψ are both assumed to be $\frac{\pi}{12}$.

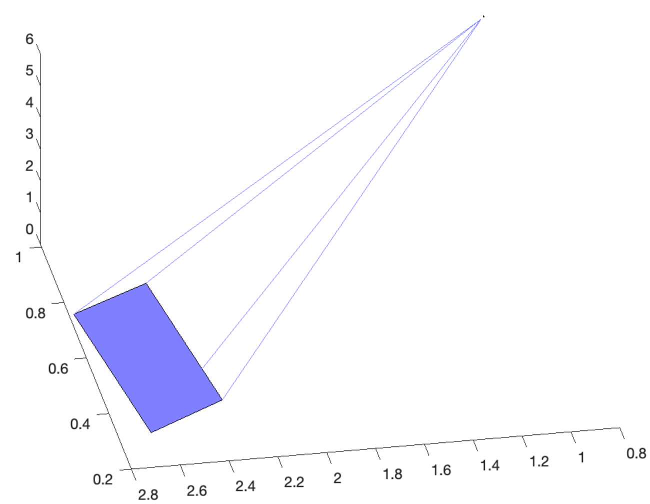


Figure 3: Field of view example

Where the blue plane shows its field of view on the bottom surface. The blue lines are the bound of the field of view.

4.2 Control system simulation

The camera kinematic parameters are listed below:

Parameter	Description	Value(units)
b_1	Motor coefficient	$100(\circ/Vs^2)$
b_2	Motor coefficient	$100(\circ/Vs^2)$
$\dot{\psi}_m$	Maximum pan angular velocity	$100(\circ/s)$
$\dot{\phi}_m$	Maximum tilt angular velocity	$100(\circ/s)$

Table 2: Information of camera

To better simulate the dynamic system, we built a simulation environment in Simulink. The system block diagram is illustrated in Figure 4.

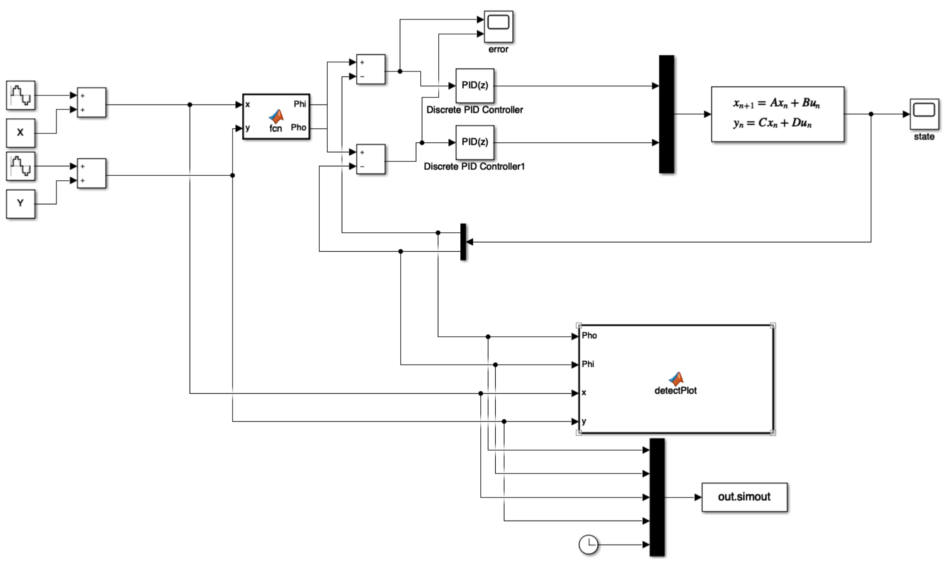


Figure 4: System in Simulink

Where I used state space to represent the camera kinematic system. Two PID controllers are used to control pan and tilt motor. In this project, the goal is to track the target. In other words, the camera will try to keep the target in the center of its field of view. A matlab function is used to calculate the optimal pan and tilt angles such that the target is at the center of FOV. And another function is used to plot the target and FOV in the 3D workspace. Please run 'callSim.m' to start the simulation.

5 Results

5.1 Testing scenario 1

In the first scenario, the target moves in a uniform circular motion with the point (1,1) as the center of the circle. The angular velocity of the target is 0.5 rad/s. The duration of the scenario is 8 seconds. Figure 5 shows the field of view and the target location during the process. An animation which shows the whole process is attached as 'scerario1.mov'

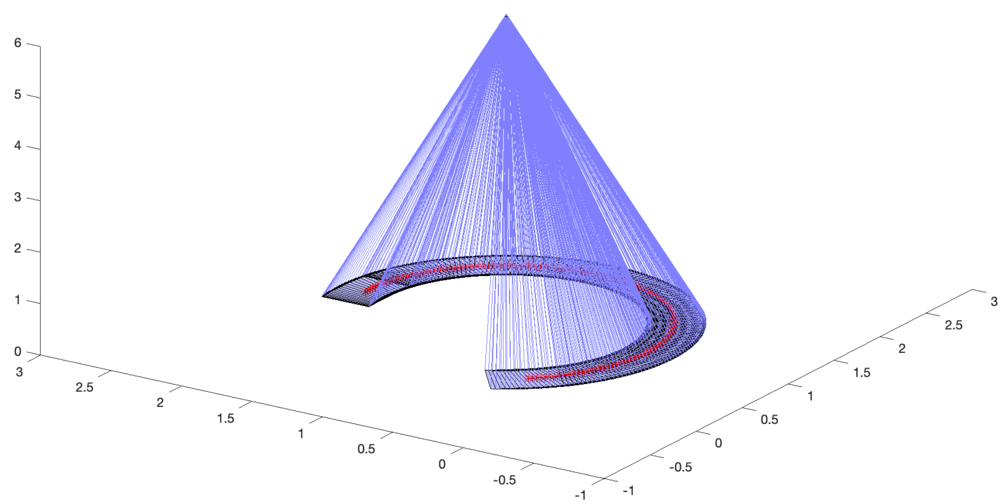


Figure 5: FOV and target

To clearly illustrate the moving process of FOV and target, I constructed a model where x and y axis represent the location of FOV and target on the bottom. The z axis represent the time of that frame. In other words, while cutting the 3D model perpendicular to the z-axis, we can intuitively see the FOV and target at a certain time point. The FOV and target movement is shown in Figure 8.

The field of view on the bottom is represented by the blue plane and the target is shown by red star. To make the figure clear, I selected only 32 frames during the process.

Figure 7a shows the states during the process. Since the target moves in a uniform circular motion around the camera, the pan angle during the process doesn't change. Due

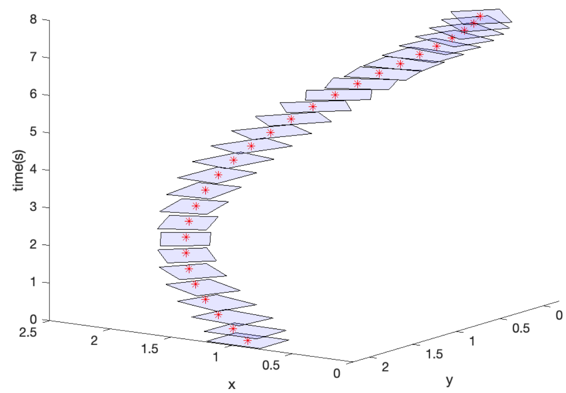
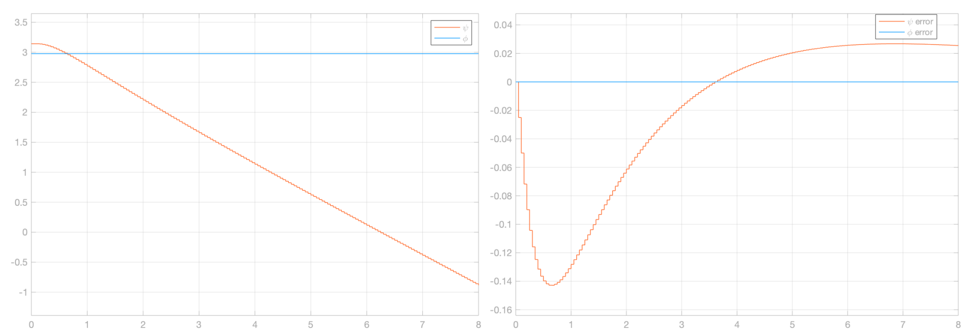


Figure 6: FOV and target at every time point

to the characteristics of the PID controller, there is a delay of the tilt angular movement at the beginning. Figure 7b illustrates the error between current pan and tilt angle and the desired angles to capture the target in the center of FOV.



(a) State (b) Error

Figure 7: Data in the simulation process

The angular speed of pan and tilt angle is shown below (in rad/s). It is within the constrain.

5.2 Testing scenario 2

In the second scenario, The target moves along the y-axis with uniform linear motion parallel to the Y axis. The speed of the target is 0.2 m/s. The duration of the scenario is 8

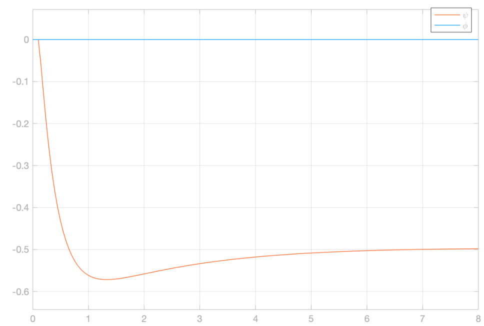


Figure 8: angular speed

seconds. Figure 9 shows the field of view and the target location during the process. An animation which shows the whole process is attached as 'scerario2.mov'

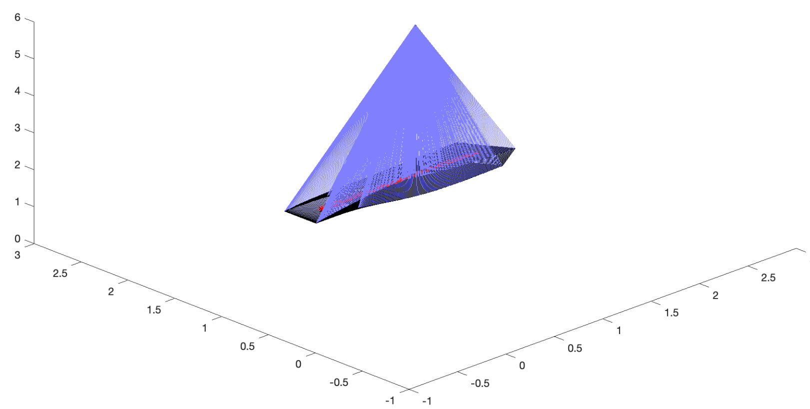


Figure 9: FOV and target

Using the same method as Figure 8. The FOV and target movement at every time point is shown in Figure 10.

Figure 11a shows the states during the process. The pan and tilt angle should change over time in this scenario. Figure 11b illustrates the error between current pan and tilt angle and the desired angles to capture the target in the center of FOV. The error is affordable during both scenarios

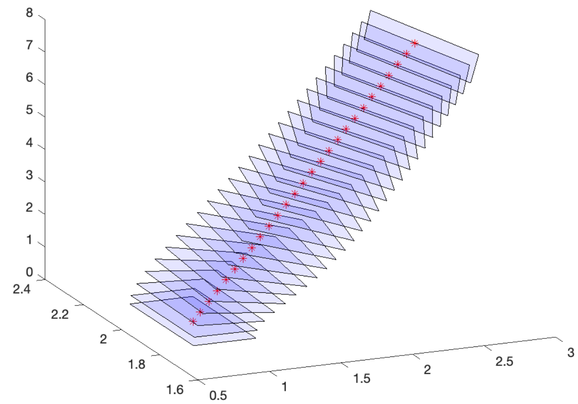
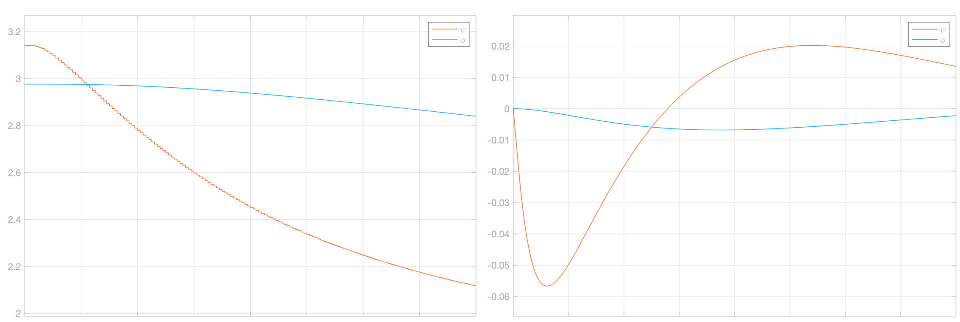


Figure 10: FOV and target at every time point



(a) State

(b) Error

Figure 11: Data in the simulation process

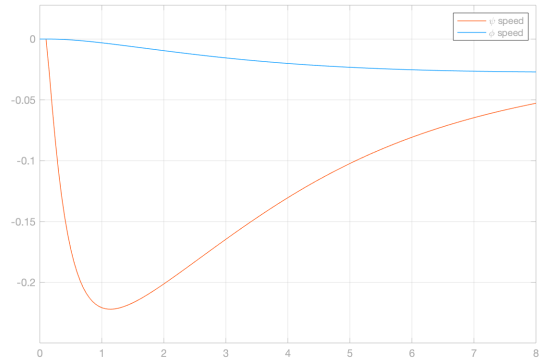


Figure 12: angular speed

The angular speed of pan and tilt angle is shown in Figure 12 (in rad/s). It is within the constrain.

References

[1] L. Ortiz, L. GonÃ§alves, and E. Cabrera, “A generic approach for error estimation of depth data from (stereo and rgb-d) 3d sensors,” 05 2017.

[2] S. Ferrari and T. Wettergren, *Information-Driven Planning and Control*, ser. Cyber Physical Systems Series. MIT Press, 2021. [Online]. Available: https://books.google.com/books?id=6nr7DwAAQBAJ